BEST AVAILABLE IMAGES

Defective images within this document are accurate representations of the original documents submitted by the applicant.

Defects in the images may include (but are not limited to):

- BLACK BORDERS
- TEXT CUT OFF AT TOP, BOTTOM OR SIDES
- FADED TEXT
- ILLEGIBLE TEXT
- SKEWED/SLANTED IMAGES
- COLORED PHOTOS
- BLACK OR VERY BLACK AND WHITE DARK PHOTOS
- GRAY SCALE DOCUMENTS

IMAGES ARE BEST AVAILABLE COPY.

As rescanning documents *will not* correct images, please do not report the images to the Image Problem Mailbox.

FIG. 1

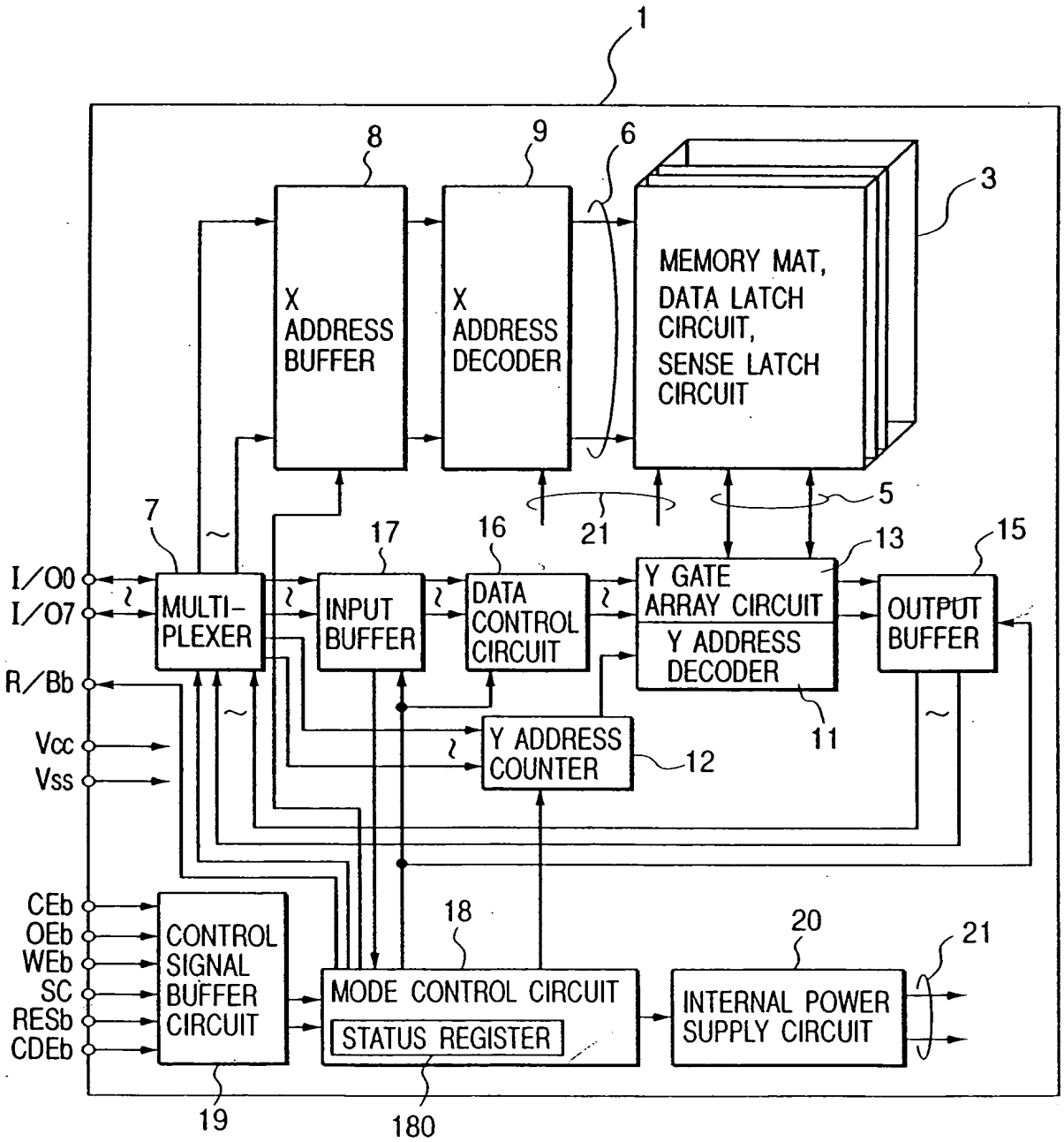

1
8
9
6
3
X ADDRESS BUFFER
X ADDRESS DECODER
MEMORY MAT, DATA LATCH CIRCUIT, SENSE LATCH CIRCUIT
7
17
16
21
5
13
15
I/O0
I/O7
MULTI-PLEXER
INPUT BUFFER
DATA CONTROL CIRCUIT
Y GATE ARRAY CIRCUIT
Y ADDRESS DECODER
OUTPUT BUFFER
R/Bb
Vcc
Vss
Y ADDRESS COUNTER
12
11
CEb
OEb
WEb
SC
RESb
CDEb
CONTROL SIGNAL BUFFER CIRCUIT
18
MODE CONTROL CIRCUIT
STATUS REGISTER
20
INTERNAL POWER SUPPLY CIRCUIT
21
19
180

FIG. 2

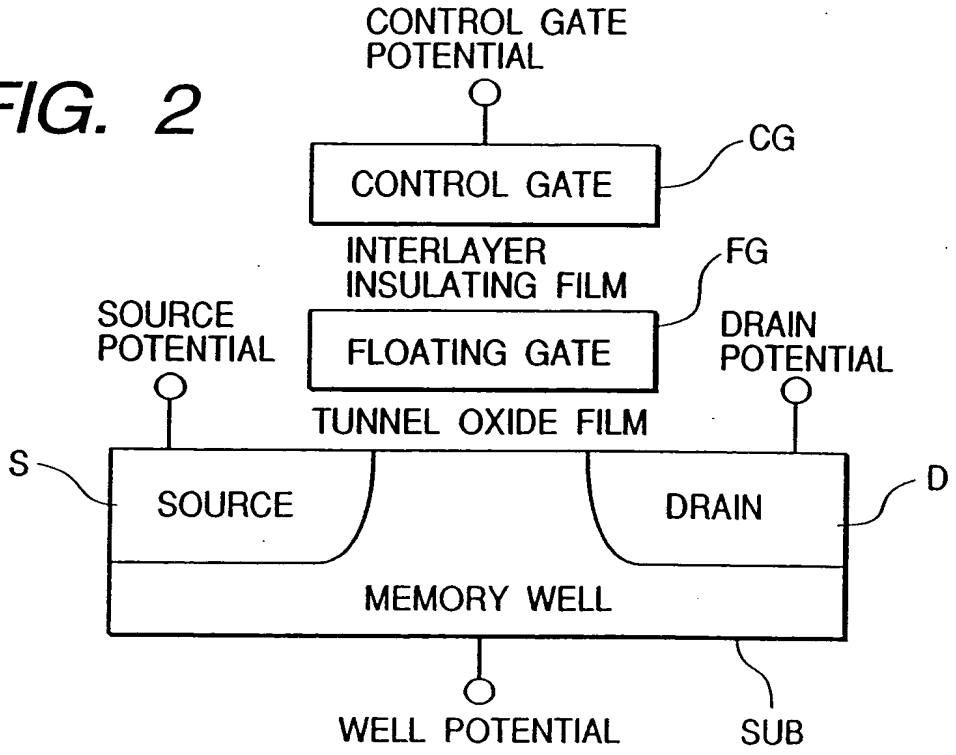


FIG. 3

MODE	FIRST COMMAND	SECOND COMMAND
READ	00H	NO NEED
RECOVERY READ	01H	NO NEED
ERASE	20H	B0H
PROGRAM	1FH	40H
ADDITIONAL PROGRAM	10H	40H
RETRY PROGRAM	1AH	NO NEED
PARTIAL ERASE	2FH	B0H
REWRITE	11H	40H

FIG. 4

	TITLE	DEFINITION
I/O7	Ready/$\overline{\text{Busy}}$	"V_{OH}"=Ready "V_{OL}"=Busy
I/O6	Reserved	
I/O5	Erase Check	"V_{OH}"=Fail "V_{OL}"=Pass
I/O4	Program Check	"V_{OH}"=Fail "V_{OL}"=Pass
I/O3	Reserved	
I/O2	Reserved	
I/O1	Reserved	
I/O0	Reserved	

STATUS REGISTER

FIG. 5

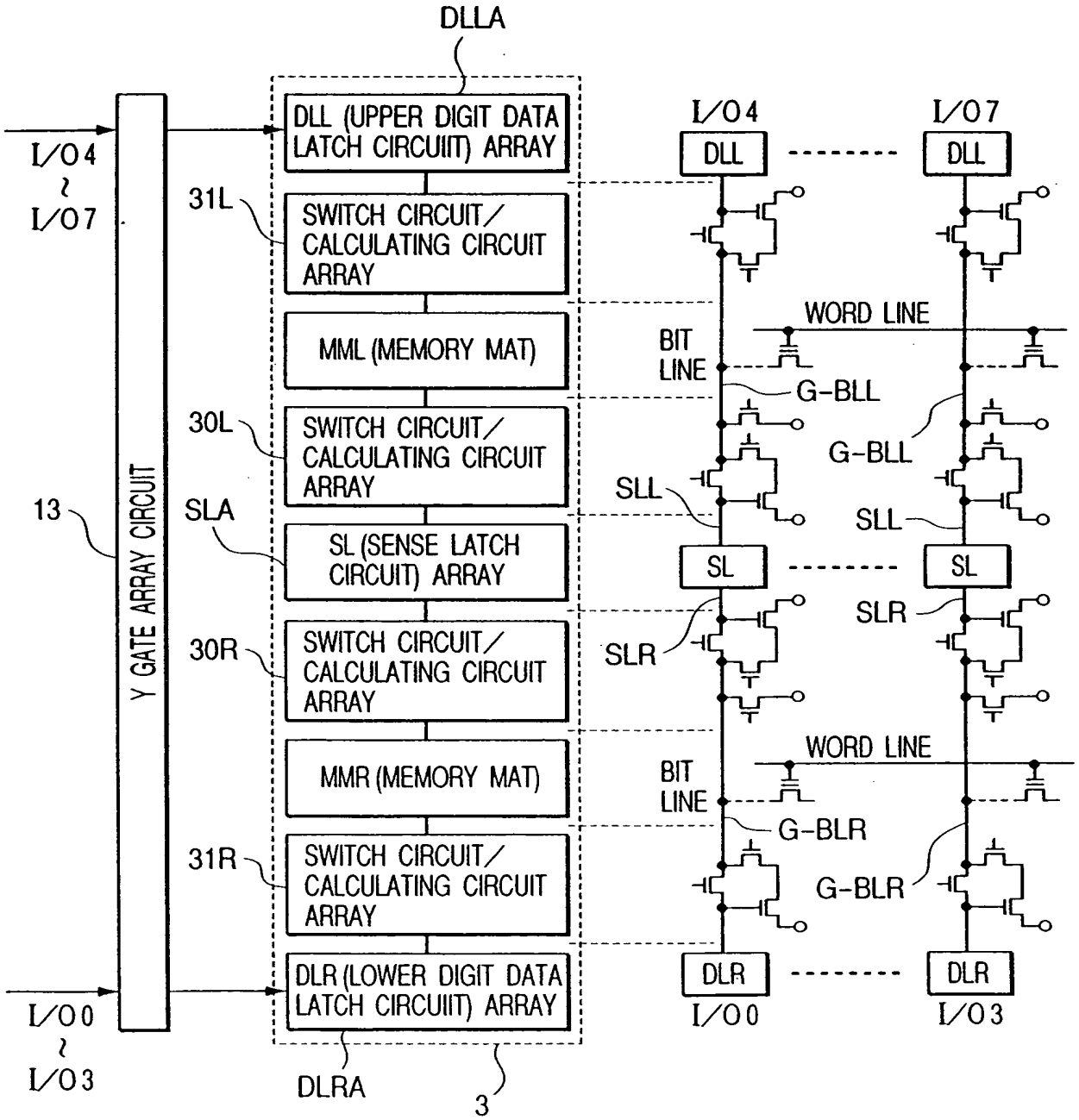

DLLA
DLL (UPPER DIGIT DATA LATCH CIRCUIIT) ARRAY
31L
SWITCH CIRCUIT/ CALCULATING CIRCUIT ARRAY
MML (MEMORY MAT)
30L
SWITCH CIRCUIT/ CALCULATING CIRCUIT ARRAY
13
SLA
SL (SENSE LATCH CIRCUIT) ARRAY
30R
SWITCH CIRCUIT/ CALCULATING CIRCUIT ARRAY
MMR (MEMORY MAT)
31R
SWITCH CIRCUIT/ CALCULATING CIRCUIT ARRAY
DLR (LOWER DIGIT DATA LATCH CIRCUIIT) ARRAY
DLRA
3
Y GATE ARRAY CIRCUIT
I/O4 ~ I/O7
I/O0 ~ I/O3
I/O4
I/O7
DLL
DLL
WORD LINE
BIT LINE
G-BLL
G-BLL
SLL
SLL
SL
SL
SLR
SLR
WORD LINE
BIT LINE
G-BLR
G-BLR
DLR
DLR
I/O0
I/O3

FIG. 6

PROGRAM DATA	I/O		DLL	DLR
	4	0		
01	0	1	0	1
00	0	0	0	0
10	1	0	1	0
11	1	1	1	1

INPUT PROGRAM DATA

FIG. 7

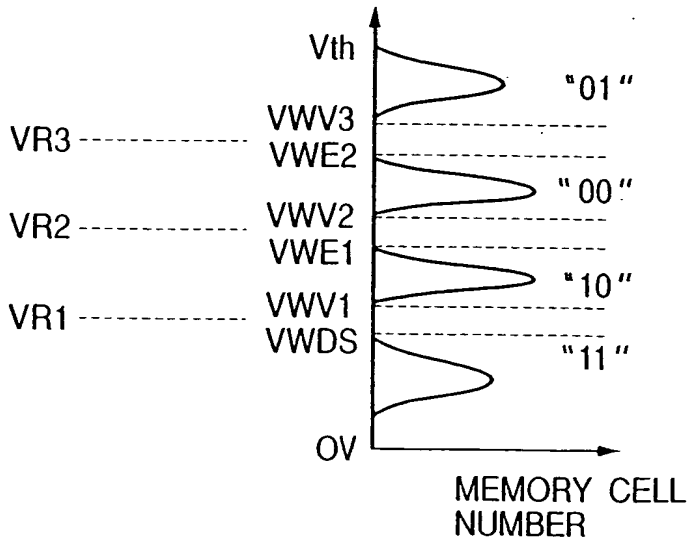
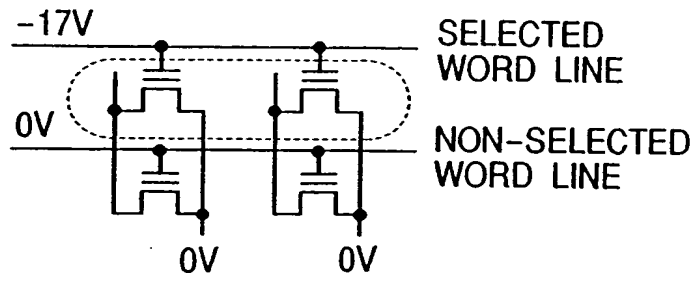
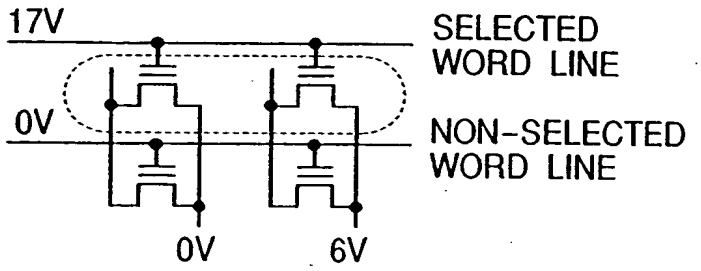
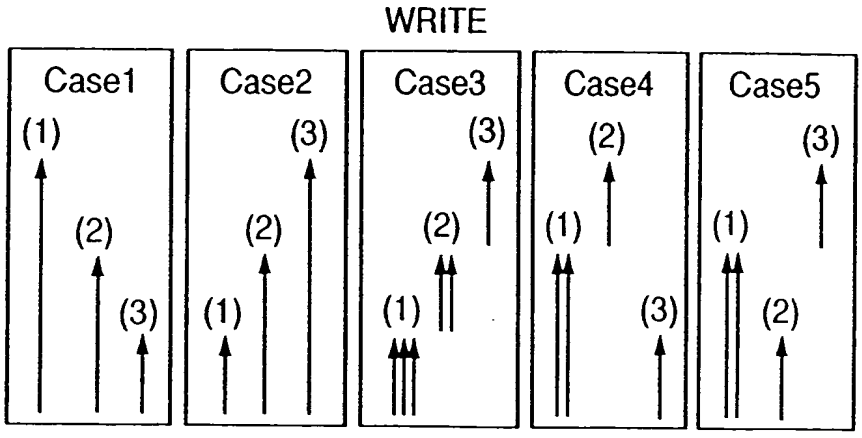


FIG. 8

SECTOR BATCH ERASE (1 WORD ERASE AT SAME TIME)	
WRITE EVERY BIT (1 WORD WRITE AT SAME TIME)	

OPERATION VOLTAGE

FIG. 9



ERASING STATE

FIG. 10

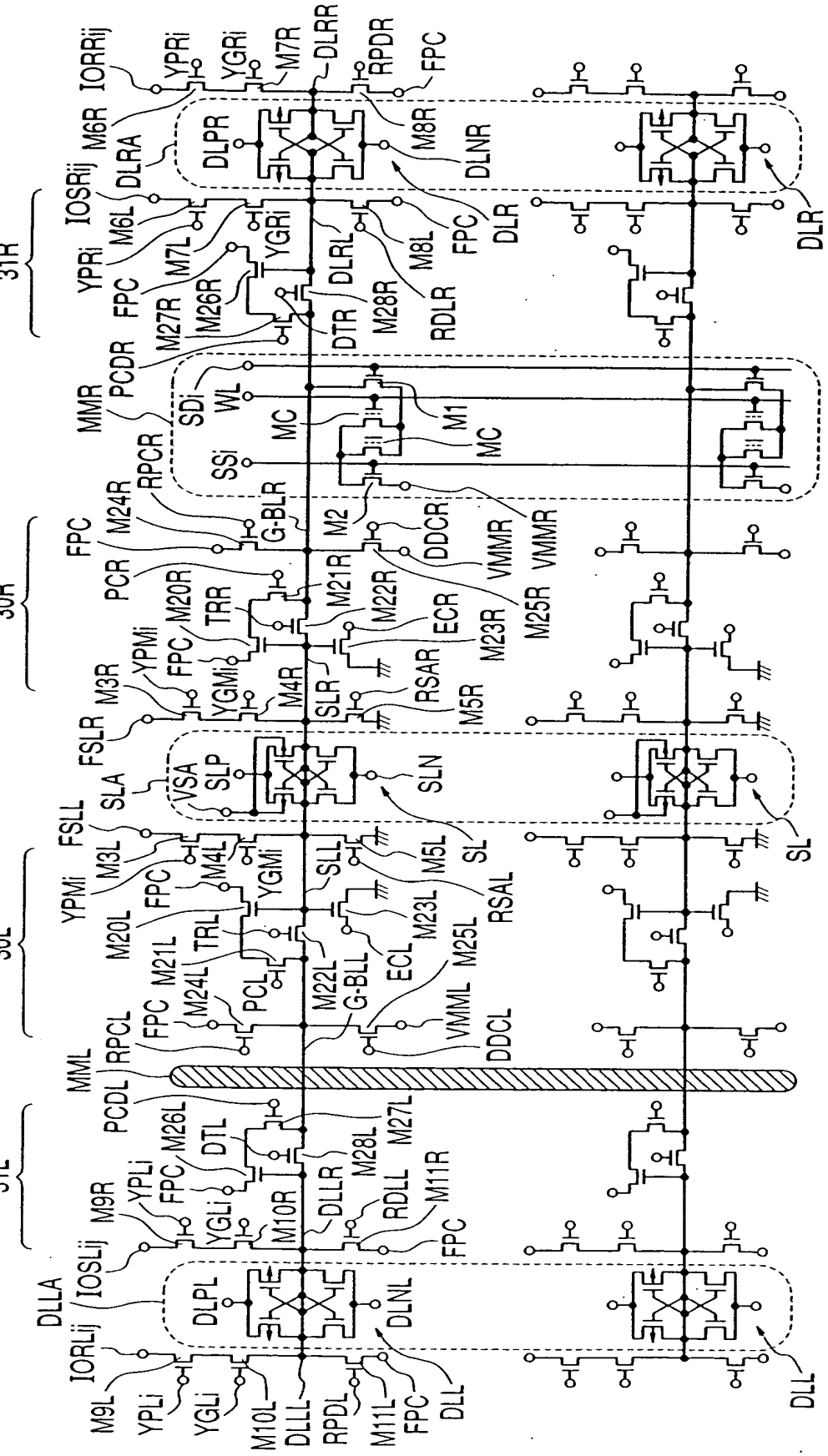

31L
30L
30R
31R
MML
MMR
DLLA
DLLR
DLPL
DLPR
DLNL
DLNR
DLL
DLR
SL
SLA
SLN
SLP
VSA
FSLL
FSLR
RPCL
RPCR
PCDL
PCDR
RPDL
RPDR
FPC
M9L
M10L
M11L
M20L
M21L
M22L
M23L
M24L
M25L
M26L
M27L
M28L
M3L
M4L
M5L
M6L
M7L
M8L
M9R
M10R
M11R
M20R
M21R
M22R
M23R
M24R
M25R
M26R
M27R
M28R
M1
M2
M3R
M4R
M5R
M6R
M7R
M8R
IOSLij
IORLij
IOSRij
IORRij
YPLi
YGLi
YPMi
YGMi
YPRi
YGRi
MC
WL
SDi
SSi
G-BLL
G-BLR
PCL
PCR
TRL
TRR
ECL
ECR
SLL
SLR
RSAL
RSAR
DTL
DTR
DLLR
DLRL
RDLL
RDLR
VMML
VMMR
DDCL
DDCR

FIG. 11

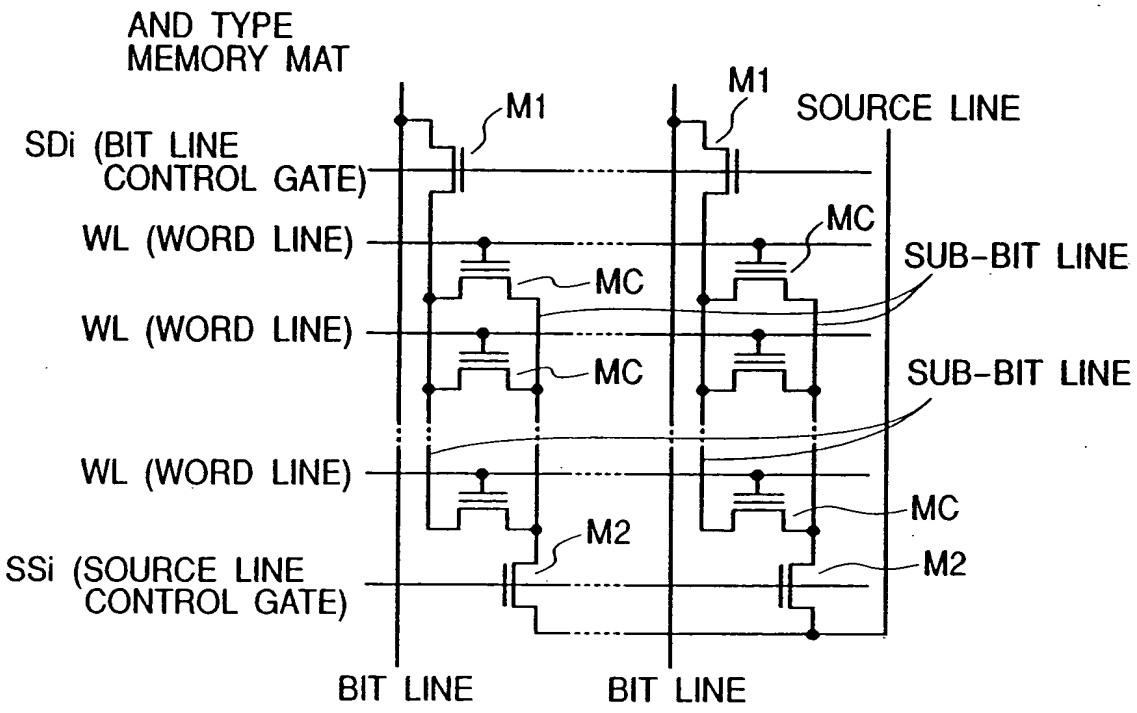

AND TYPE
MEMORY MAT
M1
M1
SOURCE LINE
SDi (BIT LINE
CONTROL GATE)
WL (WORD LINE)
MC
MC
SUB-BIT LINE
WL (WORD LINE)
MC
SUB-BIT LINE
WL (WORD LINE)
MC
M2
SSi (SOURCE LINE
CONTROL GATE)
M2
BIT LINE
BIT LINE

FIG. 12

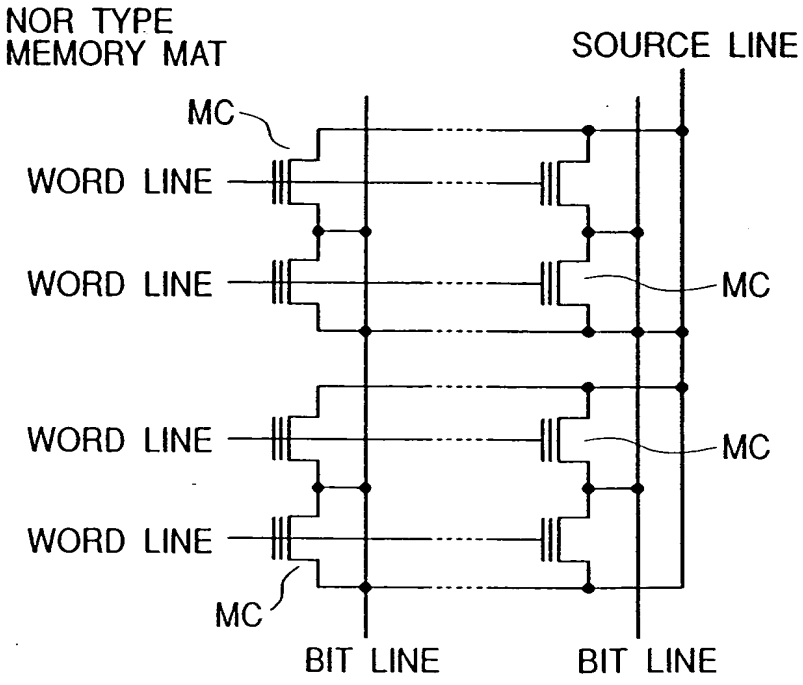

NOR TYPE
MEMORY MAT
SOURCE LINE
MC
WORD LINE
WORD LINE
MC
WORD LINE
MC
WORD LINE
MC
BIT LINE
BIT LINE

FIG. 13

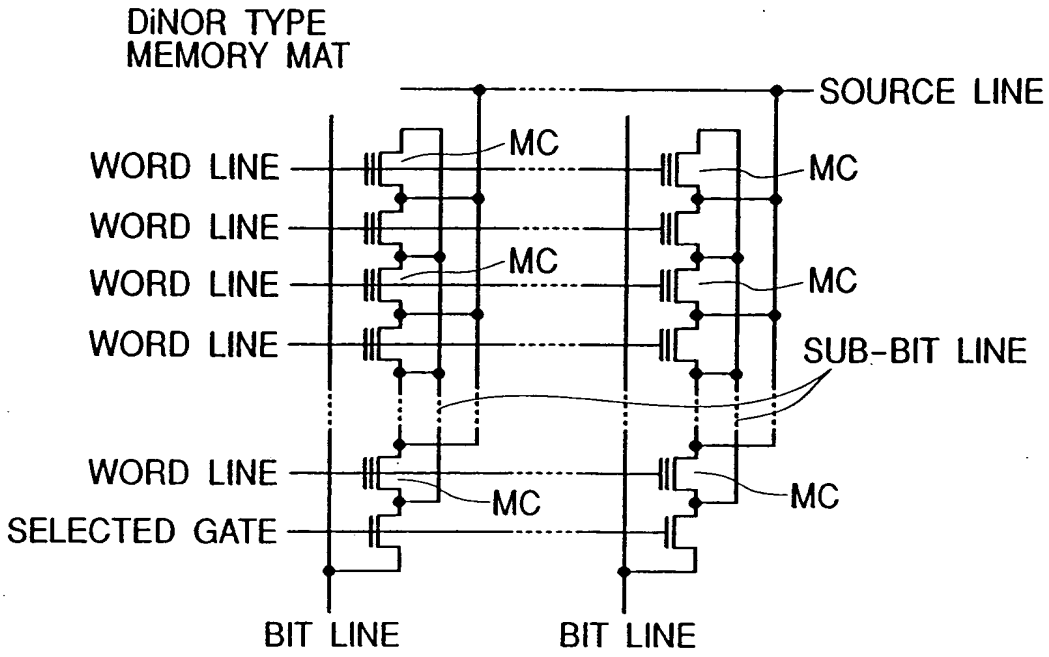

DiNOR TYPE
MEMORY MAT
SOURCE LINE
WORD LINE
WORD LINE
WORD LINE
WORD LINE
WORD LINE
SELECTED GATE
MC
MC
MC
MC
MC
MC
SUB-BIT LINE
BIT LINE
BIT LINE

FIG. 14

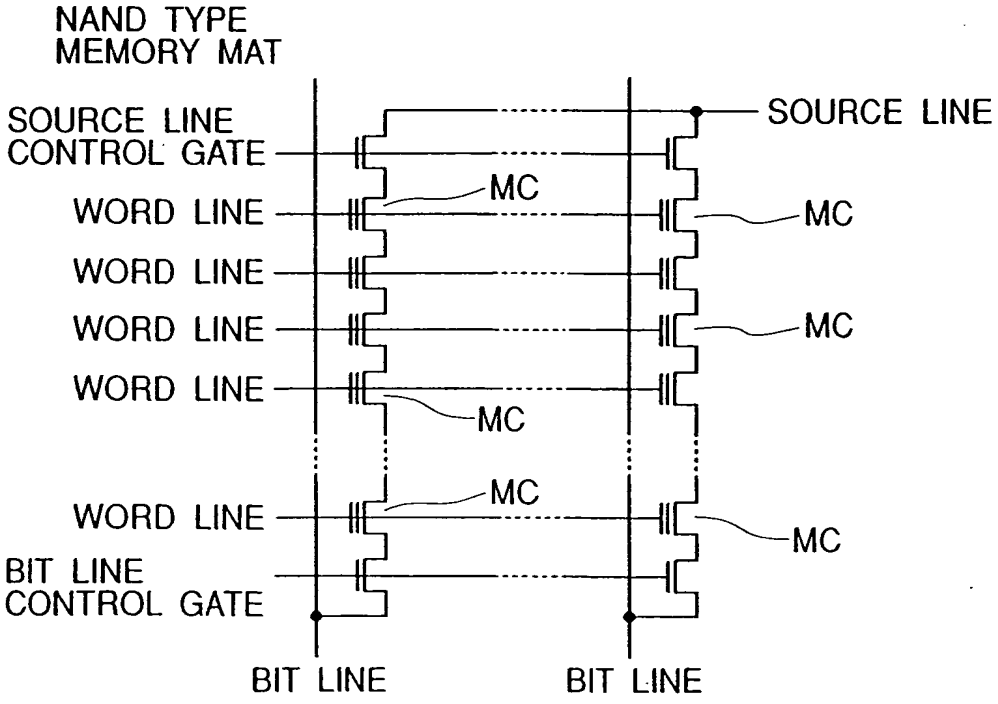

NAND TYPE
MEMORY MAT
SOURCE LINE
SOURCE LINE
CONTROL GATE
WORD LINE
WORD LINE
WORD LINE
WORD LINE
WORD LINE
BIT LINE
CONTROL GATE
MC
MC
MC
MC
MC
MC
BIT LINE
BIT LINE

FIG. 15

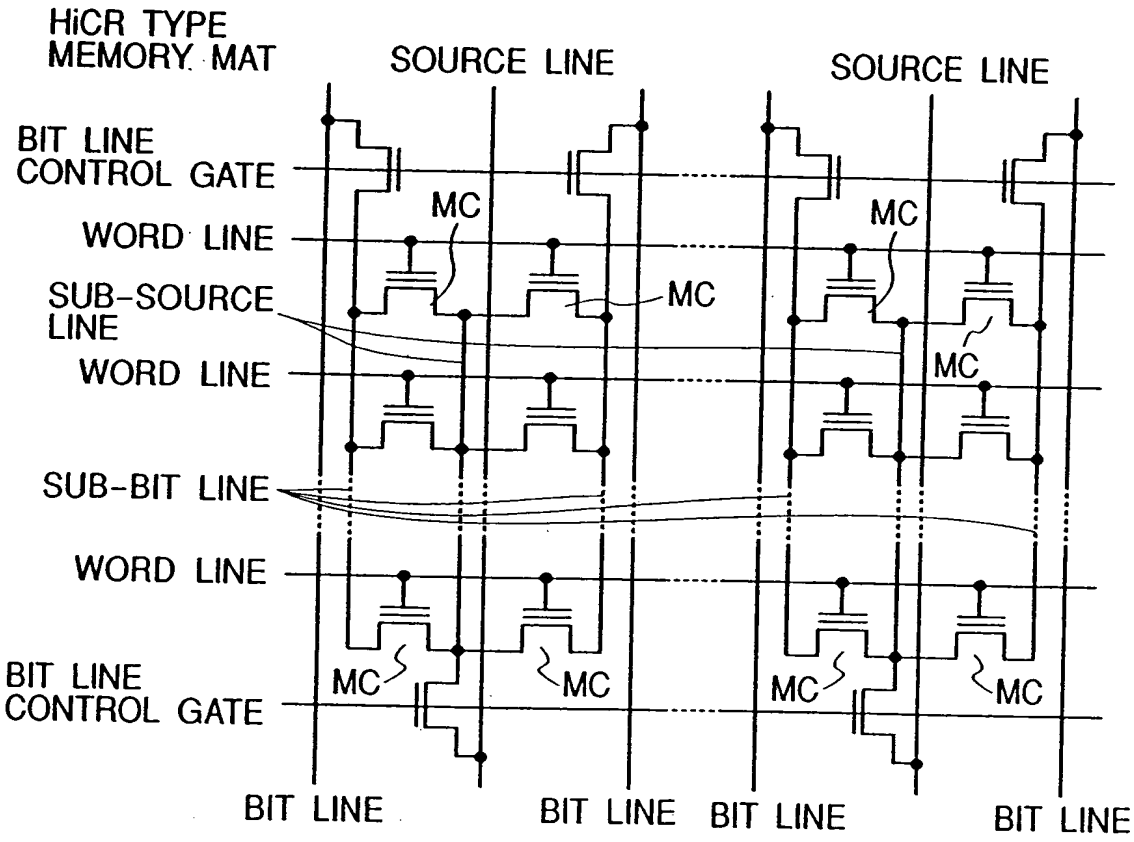
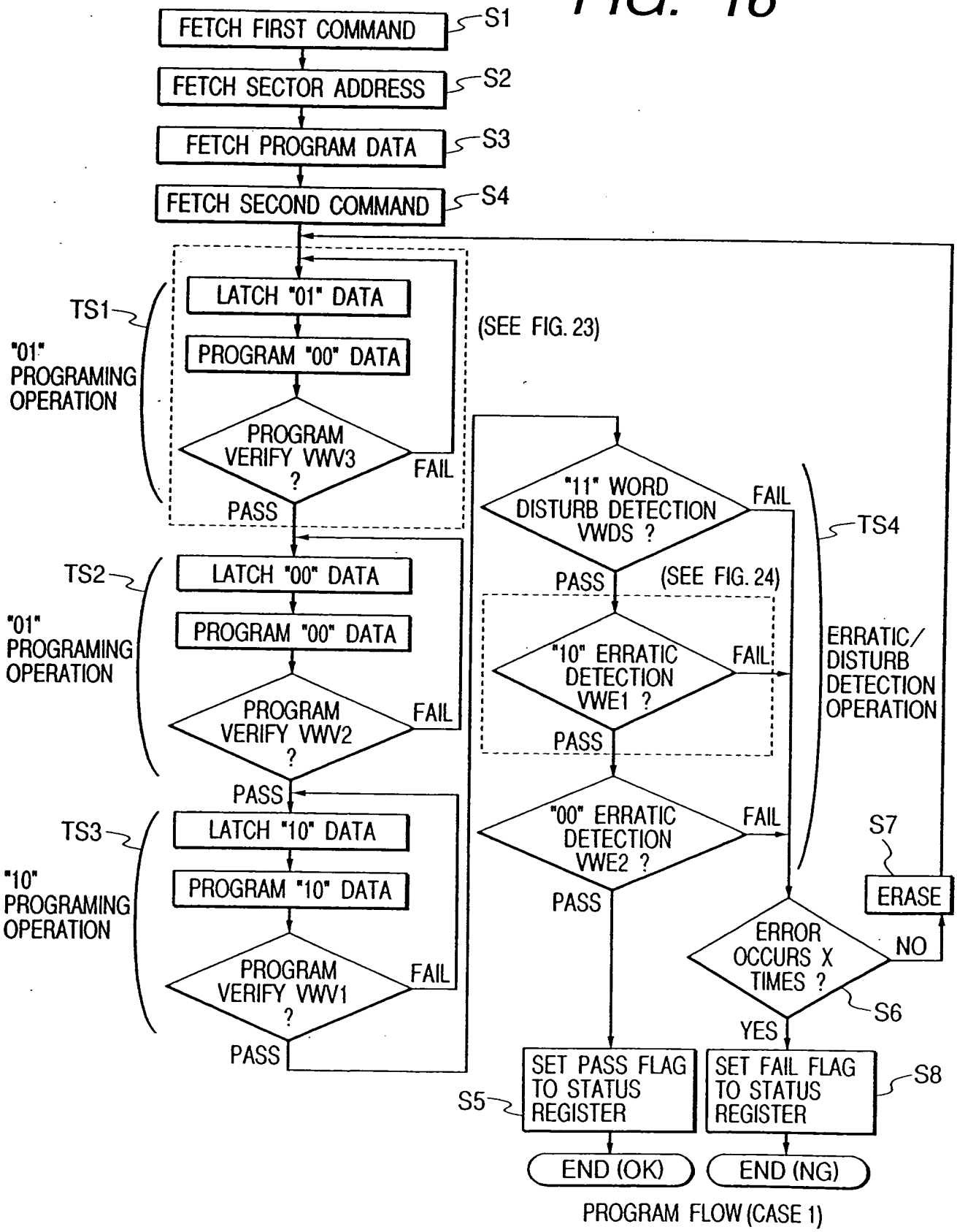

HiCR TYPE
MEMORY MAT
SOURCE LINE
SOURCE LINE
BIT LINE
CONTROL GATE
WORD LINE
SUB-SOURCE
LINE
WORD LINE
SUB-BIT LINE
WORD LINE
BIT LINE
CONTROL GATE
MC
MC
MC
MC
MC
MC
MC
MC
BIT LINE
BIT LINE
BIT LINE
BIT LINE

FIG. 16



PROGRAM FLOW (CASE 1)

FIG. 17

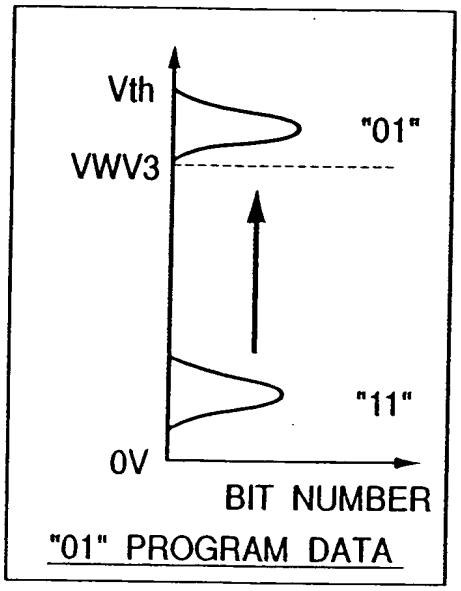

Vth
"01"
VWV3
"11"
0V
BIT NUMBER
"01" PROGRAM DATA

FIG. 18

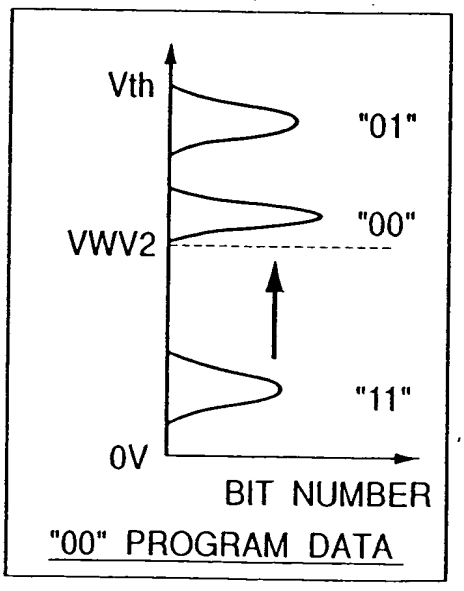

Vth
"01"
"00"
VWV2
"11"
0V
BIT NUMBER
"00" PROGRAM DATA

FIG. 19

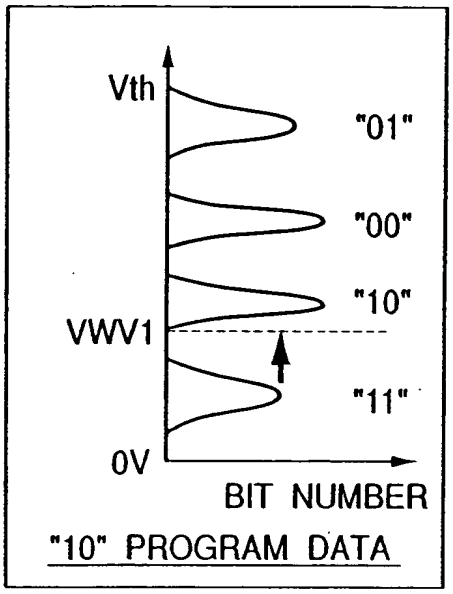

Vth
"01"
"00"
"10"
VWV1
"11"
0V
BIT NUMBER
"10" PROGRAM DATA

FIG. 20

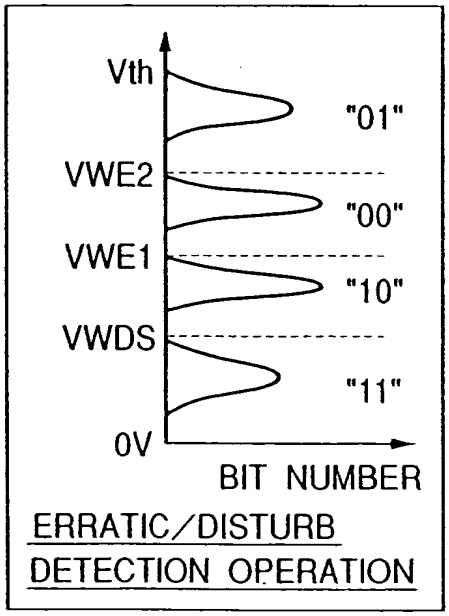

Vth
"01"
VWE2
"00"
VWE1
"10"
VWDS
"11"
0V
BIT NUMBER
ERRATIC/DISTURB
DETECTION OPERATION

FIG. 21

DATA LATCH PROCESS	CALCULATION CONTENT (SENSE LATCH DATA OF SELECTED MAT SIDE)
"01" PROGRAM DATA	$A+\overline{B}$
"00" PROGRAM DATA	$A+B$
"10" PROGRAM DATA	$\overline{A}+B$
"00" ERRATIC DETECTION DATA	$\overline{A+B}$
"10" ERRATIC DETECTION DATA	$A \cdot \overline{B}$
"11" DISTURB DETECTION DATA	$A \cdot B$

A : UPPER DIGIT DATA B : LOWER DIGIT DATA

FIG. 22

A UPPER DIGIT	B LOWER DIGIT	$A+\overline{B}$	$A+B$	$\overline{A}+B$	$\overline{A+B}$	$A \cdot \overline{B}$	$A \cdot B$
0	1	0	1	1	0	0	0
0	0	1	0	1	1	0	0
1	0	1	1	0	0	1	0
1	1	1	1	1	0	0	1

FIG. 23

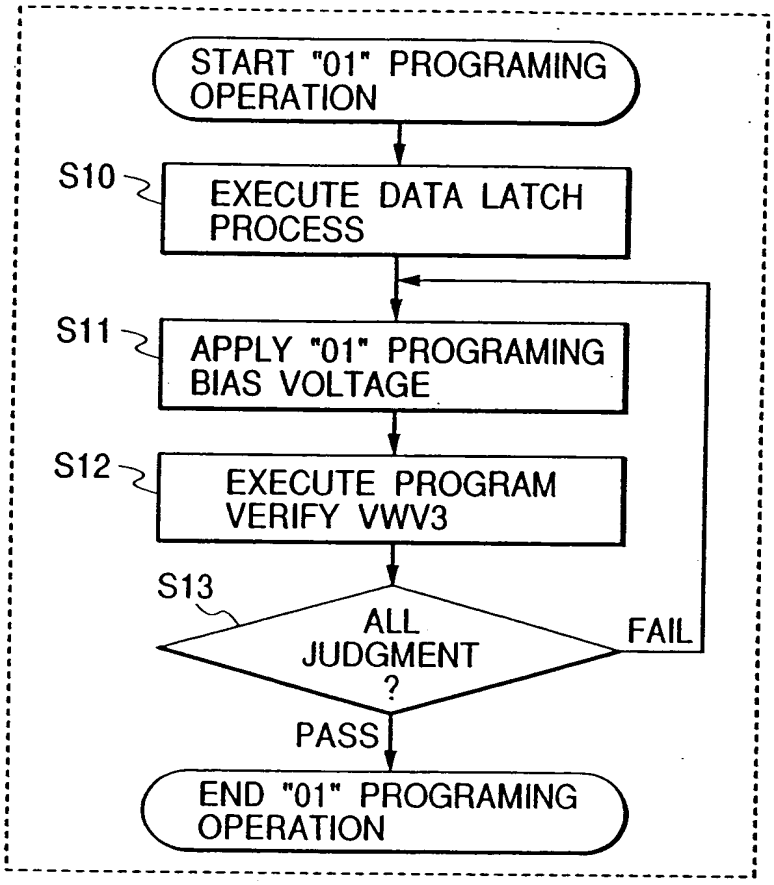

START "01" PROGRAMING OPERATION
S10
EXECUTE DATA LATCH PROCESS
S11
APPLY "01" PROGRAMING BIAS VOLTAGE
S12
EXECUTE PROGRAM VERIFY VWV3
S13
ALL JUDGMENT ?
FAIL
PASS
END "01" PROGRAMING OPERATION

FIG. 24

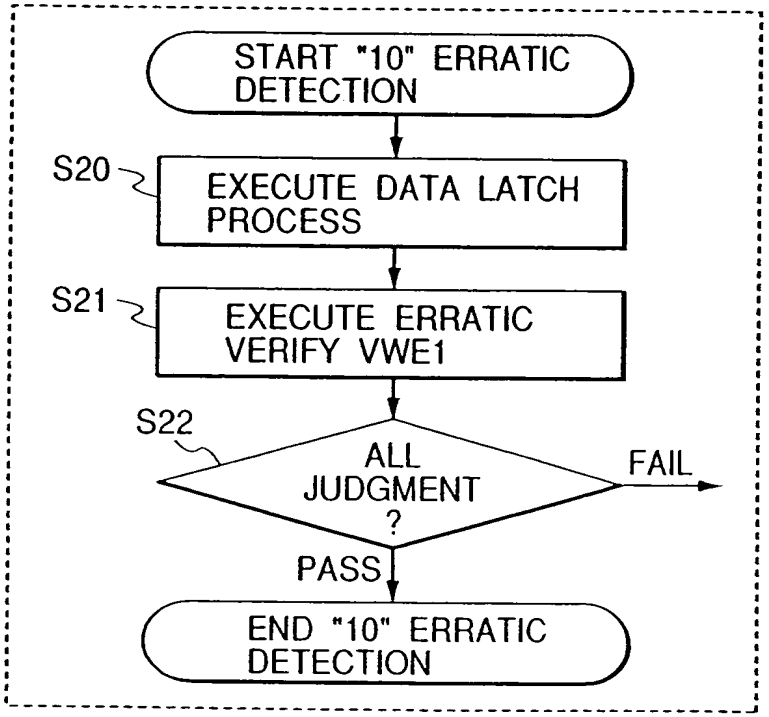
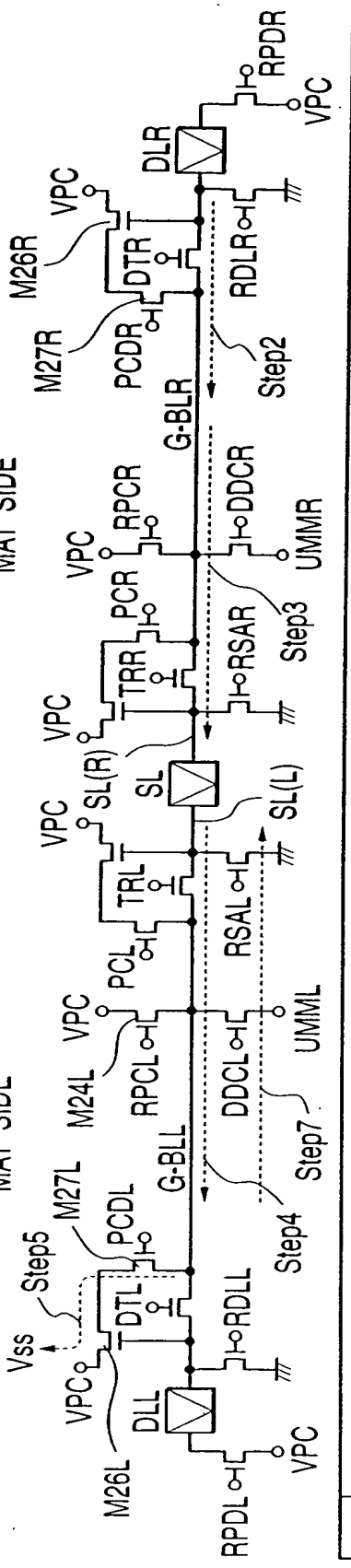

START "10" ERRATIC DETECTION
S20
EXECUTE DATA LATCH PROCESS
S21
EXECUTE ERRATIC VERIFY VWE1
S22
ALL JUDGMENT ?
FAIL
PASS
END "10" ERRATIC DETECTION

FIG. 25



Step		Step 2				Step 3				Step 4				Step 5				Step 6				Step 7			
CONTENT		DATA TRANSFER DLR → G-BLR PRECHARGE G-BLL				SL SENSE				CLEAR G-BLR/L DATA TRANSFER SL(L) → G-BLL				CALCULATE (DLL, G-BLL) CLEAR SL				PRECHARGE G-BLR				SL SENSE			
		DLL	G-BLL	SL(L)	SL(R)	DLR	G-BLR	SL(R)	SL(L)	DLL	G-BLL	SL(L)	SL(R)	DLR	G-BLR	SL(R)	SL(L)	DLL	G-BLL	SL(L)	SL(R)	DLR	G-BLR	SL(R)	SL(L)	G-BLL	DLL	DLR	G-BLR	SL(R)	SL(L)	G-BLL	DLL	DLR
01	0	0.5				0	0.0	0(1)	0	1	1.0	1	0	0	0	0	1	0	1.0	0	0	0(1)	0.5	0	1	1	0	0	0(1)	0	1	1	0	0(1)
00	0	0.5				0	1.0	1(0)	0	0	0.0	0	1	1(0)	0	0	0	0	0.0	0	0	1(0)	0.5	0	0	0	1	1	1(0)	0	0	0	1	1(0)
10	1	0.5				1	1.0	1(0)	1	0	0.0	0	1	1(0)	1	0	0	1	0.0	0	0	1(0)	0.5	1	0	0	1	1	1(0)	1	0	0	1	1(0)
11	1	0.5				0	0.0	0(1)	1	1	1.0	1	0	0(1)	1	0	0	1	0.0	0	0	0(1)	0.5	1	0	0	1	1	0(1)	1	0	0	1	0(1)

(a) (b) (c) (d) (e) (f) (g) (h) (i) (j)

PROCESSED RESULT

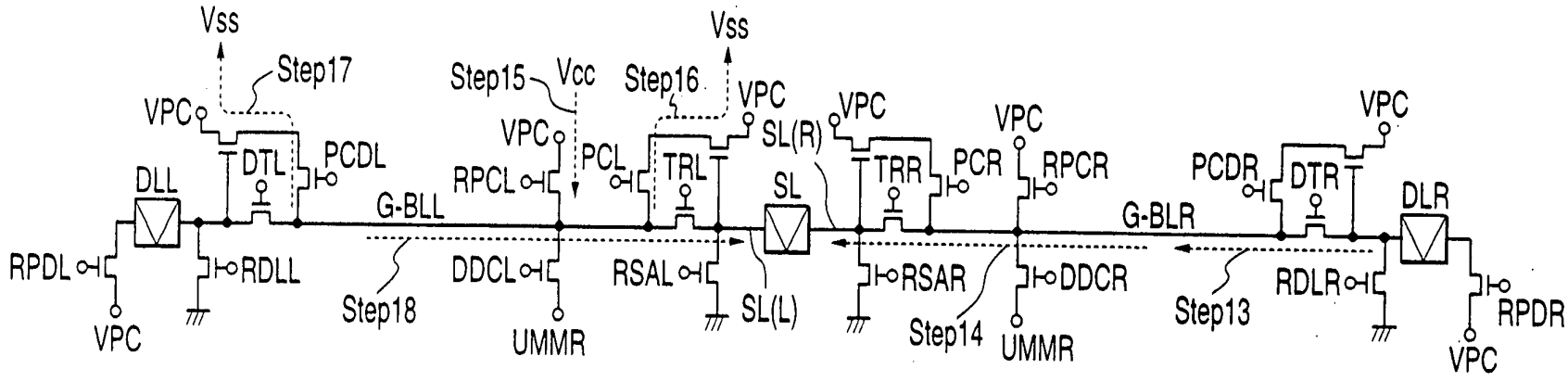
LATCH DATA TO BIT LINE SIDE OF DATA LATCH CIRCUIT ON SIDE OF NON-SELECTED MAT

LATCH DATA TO OPPOSITE SIDE/ BIT LINE OF DATA LATCH CIRCUIT ON SIDE OF SELECTED MAT IN MULTI-SENSE METHOD

- BIT LINE USED TO PPROGRAM DATA IS "0"
- BIT LINE USED NOT TO PPROGRAM DATA IS "1" DUE TO APPLICATION OF BLOCK VOLTAGE

'01' PROGRAM DATA LATCH PROCESS OPERATION (MULTI-SENSE METHOD)

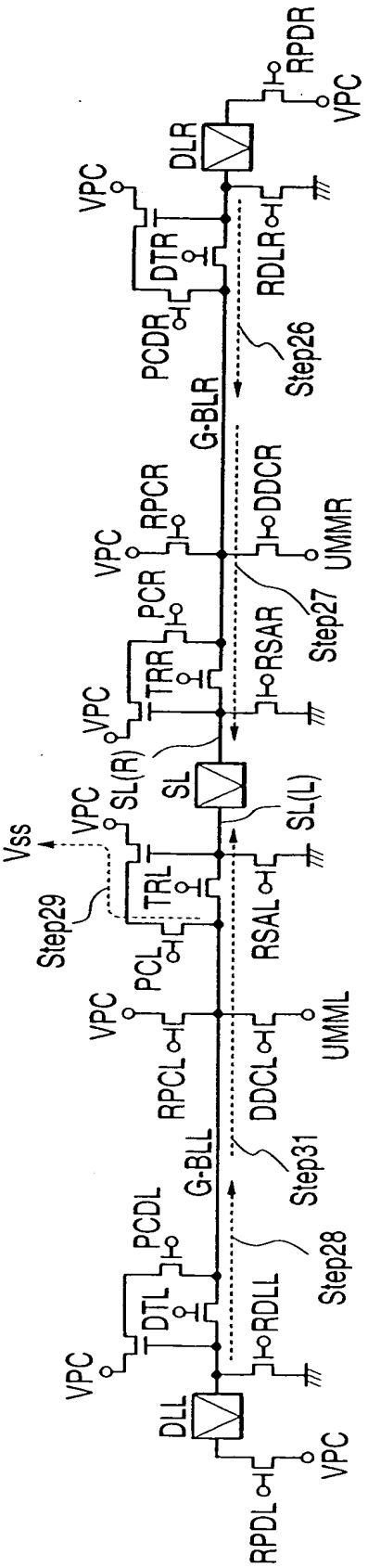
FIG. 26



Step	PROGRAM DATA LATCH																																	
	Step 13						Step 14						Step 15						Step 16						Step 17						Step 18					
CONTENT	DATA TRANSFER DLR — G-BLR PRECHARGE G-BLL						SL SENSE						CLEAR G-BLR/L PRECHARGE G-BLR/L						CALCULATE (G-BLL, SL(L))						CALCULATE (DLL, G-BLL) SL CLEAR						SL SENSE					
	DLL	G-BLL	SL(L)	SL(R)	G-BLR	DLR	DLL	G-BLL	SL(L)	SL(R)	G-BLR	DLR	DLL	G-BLL	SL(L)	SL(R)	G-BLR	DLR	DLL	G-BLL	SL(L)	SL(R)	G-BLR	DLR	DLL	G-BLL	SL(L)	SL(R)	G-BLR	DLR	DLL	G-BLL	SL(L)	SL(R)	G-BLR	DLR
01	0	0.5			0.0	0(1)	0	1	1	0	0	0(1)	0	1.0	1	0	0.5	0(1)	0	0.0	1	0	0.5	0(1)	0	0.0	0	0	0.5	0(1)	0	0	0	1	1	0(1)
00	0	0.5			1.0	1(0)	0	0	0	1	1	1(0)	0	1.0	0	1	0.5	1(0)	0	1.0	0	1	0.5	1(0)	0	1.0	0	0	0.5	1(0)	0	1	1	0	0	1(0)
10	1	0.5			1.0	1(0)	1	0	0	1	1	1(0)	1	1.0	0	1	0.5	1(0)	1	1.0	0	1	0.5	1(0)	1	0.0	0	0	0.5	1(0)	1	0	0	1	1	1(0)
11	1	0.5			0.0	0(1)	1	1	1	0	0	0(1)	1	1.0	1	0	0.5	0(1)	1	0.0	1	0	0.5	0(1)	1	0.0	0	0	0.5	0(1)	1	0	0	1	1	0(1)

'00' PROGRAM DATA LATCH PROCESS OPERATION (MULTI-SENSE METHOD)

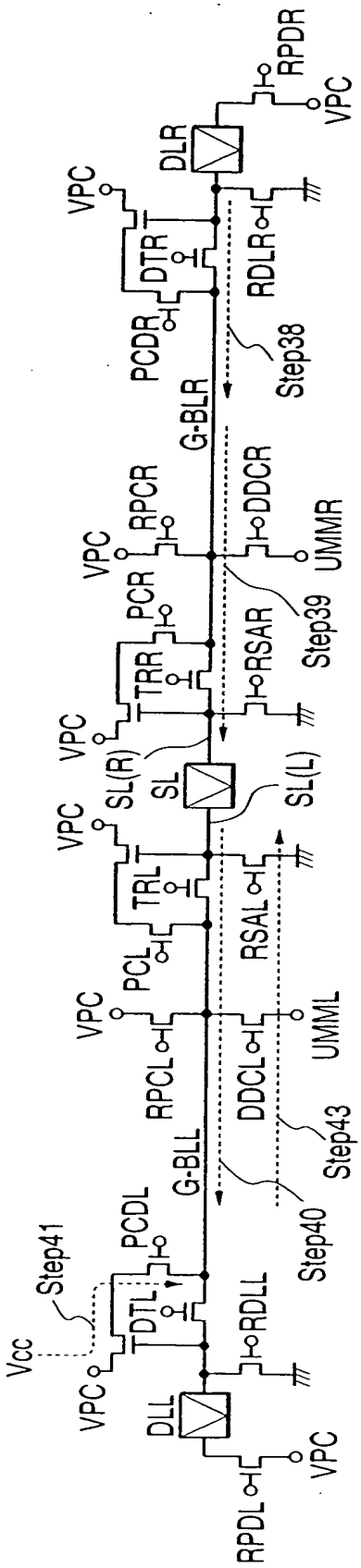
FIG. 27



Step		PROGRAM DATA LATCH																													
		Step 26					Step 27					Step 28					Step 29					Step 30					Step 31				
CONTENT		DATA TRANSFER DLR — G-BLR PRECHARGE G-BLL					SL SENSE					CLEAR G-BLR/L DATA TRANSFER DLR — G-BLR					CALCULATE (G-BLL, SL(L))					SL CLEAR PRECHARGE G-BLR					SL SENSE				
		DLL	G-BLL	SL(L)	SL(R)	G-BLR	DLR	DLL	G-BLL	SL(L)	SL(R)	G-BLR	DLR	DLL	G-BLL	SL(L)	SL(R)	G-BLR	DLR	DLL	G-BLL	SL(L)	SL(R)	G-BLR	DLR	DLL	G-BLL	SL(L)	SL(R)	G-BLR	DLR
01	0	0	0.5			0.0	0(1)	0	1	1	0	0	0(1)	0	0.0	1	0	0	0(1)	0	0.0	0	0	0.5	0(1)	0	0	1	1	0(1)
00	0	0	0.5			1.0	1(0)	0	0	0	1	1	1(0)	0	0.0	0	1	0	1(0)	0	0.0	0	0	0.5	1(0)	0	0	1	1	1(0)
10	1	1	0.5			1.0	1(0)	1	0	0	1	1	1(0)	1	1.0	0	1	0	1(0)	1	1.0	0	0	0.5	1(0)	1	1	0	0	1(0)
11	1	1	0.5			0.0	0(1)	1	1	1	0	0	0(1)	1	0.0	1	0	0	0(1)	1	0.0	0	0	0.5	0(1)	1	0	0	1	0(1)

'10' PROGRAM DATA LATCH PROCESS OPERATION (MULTI-SENSE METHOD)

FIG. 28



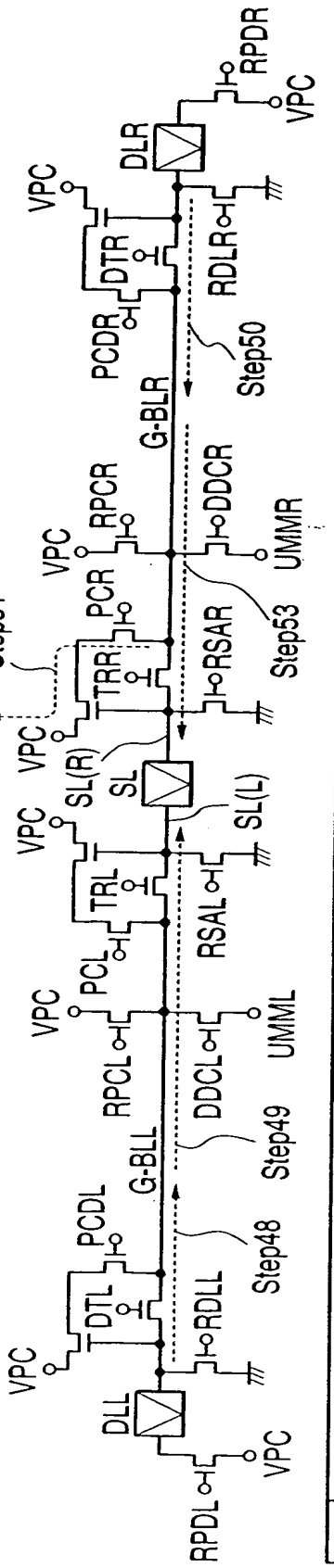
Step		ERRATIC DATA LATCH																									
		Step 38					Step 39					Step 40					Step 41					Step 42					Step 43				
CONTENT		DATA TRANSFER DLR → G-BLR PRECHARGE G-BLL					SL SENSE					CLEAR G-BLR/L DATA TRANSFER SL(L) → G-BLL					CALCULATE (DLL, G-BLL) SL CLEAR					PRECHARGE G-BLR					SL SENSE				
		DLL	G-BLL	SL(L)	SL(R)	G-BLR	DLR	DLL	G-BLL	SL(L)	SL(R)	G-BLR	DLR	DLL	G-BLL	SL(L)	SL(R)	G-BLR	DLR	DLL	G-BLL	SL(L)	SL(R)	G-BLR	DLR	DLL	G-BLL	SL(L)	SL(R)	G-BLR	DLR
01	0	0.5				0.0	0(1)	0	1.0			0	0(1)	0	1.0			0	0(1)	0	1.0	0	0	0.5	0(1)	0	1	1	0	0	0(1)
00	0	0.5				1.0	1(0)	0	0.0	0	1	0	1(0)	0	0.0	0	0	0	1(0)	0	0.0	0	0	0.5	1(0)	0	0	1	1	1	1(0)
10	1	0.5				1.0	1(0)	1	0.0	0	1	0	1(0)	1	1.0	0	0	0	1(0)	1	1.0	0	0	0.5	1(0)	1	1	1	0	0	1(0)
11	1	0.5				0.0	0(1)	1	1.0	1	0	0	0(1)	1	1.0	0	0	0	0(1)	1	1.0	0	0	0.5	0(1)	1	1	1	0	0	0(1)

'00' ERRATIC DETECTION DATA LATCH PROCESS OPERATION
(MULTI-SENSE METHOD)

← PROCESSED RESULT

ONLY BIT LINE FOR ERRATIC
DETECTION IS "1" (SELECTIVELY PRECHARGE)

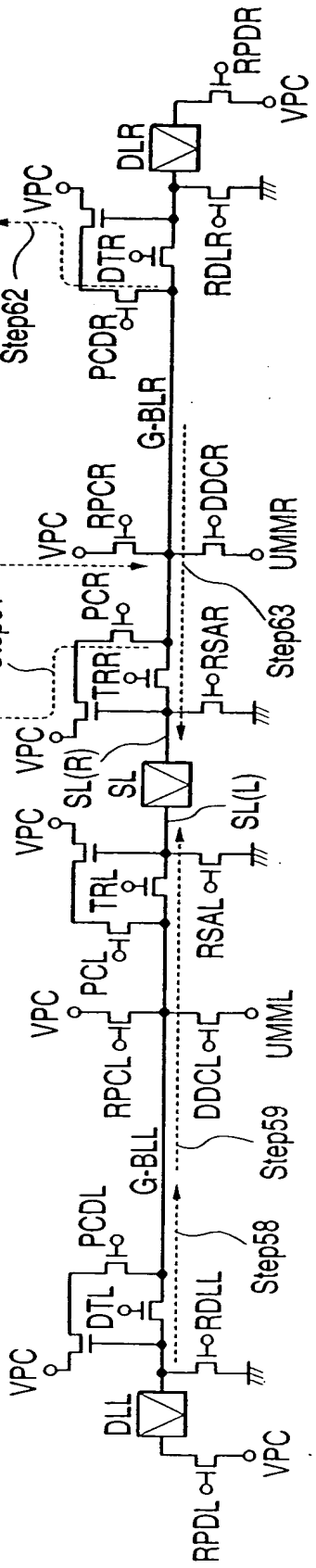
FIG. 29



Step		ERRATIC DATA LATCH																									
		Step 48				Step 49				Step 50				Step 51				Step 52				Step 53					
CONTENT		DATA TRANSFER DLL — G-BLL PRECHARGE G-BLR				SL SENSE CLEAR G-BLR/L				DATA TRANSFER DLR — G-BLR				CALCULATE (SL(R), G-BLR)				SL CLEAR PRECHARGE G-BLL				SL SENSE					
		DLL	G-BLL	SL(L)	SL(R)	DLL	G-BLL	SL(L)	SL(R)	DLL	G-BLL	SL(L)	SL(R)	DLL	G-BLL	SL(L)	SL(R)	DLL	G-BLL	SL(L)	SL(R)	DLL	G-BLL	SL(L)	SL(R)	G-BLR	DLR
01	0	0.0			0.5	0	0	0	1	0	0	0	1	0	0	1	0.0	0(1)	0	0.5	0	0	0.0	0(1)	0	1	0	0(1)
00	0	0.0			0.5	0	0	0	1	0	0	0	1	0	0	1	0.0	1(0)	0	0.5	0	0	0.0	1(0)	0	1	0	1(0)
10	1	1.0			0.5	1	0	0	1	1	0	0	1	1	0	1	1.0	1(0)	1	0.5	0	0	1.0	1(0)	1	0	1	1(0)
11	1	1.0			0.5	0(1)	1	0	1	0(1)	1	0	1	0(1)	1	0	0.0	0(1)	1	0.5	0	0	0.0	0(1)	1	1	0	0(1)

'10' ERRATIC DETECTION DATA LATCH PROCESS OPERATION (MULTI-SENSE METHOD)

FIG. 30



Step		DISTURB DATA LATCH																													
		Step 58				Step 59				Step 60				Step 61				Step 62				Step 63									
CONTENT		DATA TRANSFER DLL → G-BLL PRECHARGE G-BLR				SL SENSE CLEAR G-BLR/L				PRECHARGE G-BLR/L				CALCULATE (SL(R), G-BLR)				CALCULATE (G-BLR, DLR) SL CLEAR				SL SENSE									
		DLL	G-BLL	SL(L)	SL(R)	G-BLR	DLR	DLL	G-BLL	SL(L)	SL(R)	G-BLR	DLR	DLL	G-BLL	SL(L)	SL(R)	G-BLR	DLR	DLL	G-BLL	SL(L)	SL(R)	G-BLR	DLR	DLL	G-BLL	SL(L)	SL(R)	G-BLR	DLR
01	0	0	0.0			0.5	0(1)	0	0.5	0	1	1.0	0(1)	0	0.5	0	1	0.0	0(1)	0	0.5	0	0	0.0	0(1)	0	1	1	0	0	0(1)
00	0	0	0.0			0.5	1(0)	0	0.5	0	1	1.0	1(0)	0	0.5	0	1	0.0	1(0)	0	0.5	0	0	0.0	1(0)	0	1	1	0	0	1(0)
10	1	1	1.0			0.5	1(0)	1	0.5	1	0	1.0	1(0)	1	0.5	1	0	1.0	1(0)	1	0.5	0	0	0.0	1(0)	1	1	1	0	0	1(0)
11	1	1	1.0			0.5	0(1)	1	0.5	1	0	1.0	0(1)	1	0.5	1	0	1.0	0(1)	1	0.5	0	0	1.0	0(1)	1	0	0	1	1	0(1)

'11' DISTURB DETECTION DATA LATCH PROCESS OPERATION (MULTI-SENSE METHOD)

FIG. 31

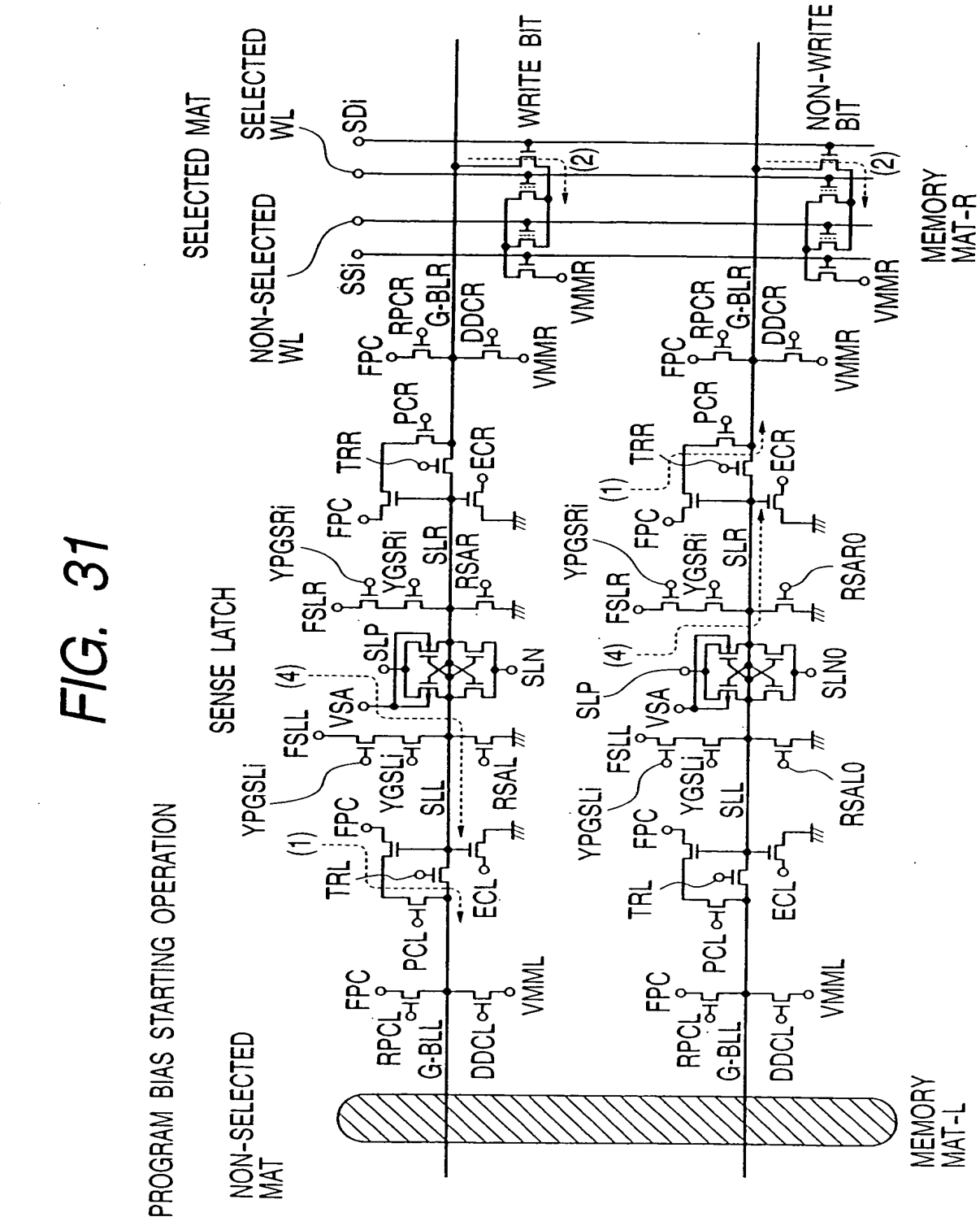

PROGRAM BIAS STARTING OPERATION
NON-SELECTED MAT
SENSE LATCH
SELECTED MAT
NON-SELECTED WL
SELECTED WL
WRITE BIT
NON-WRITE BIT
MEMORY MAT-L
MEMORY MAT-R
YPGSLi
FSLL
VSA
SLP
SLN
SLN0
RSAL
RSAL0
YGSLi
SLL
TRL
FPC
PCL
RPCL
G-BLL
DDCL
ECL
VMML
YPGSRi
FSLR
YGSRi
SLR
RSAR
RSAR0
TRR
PCR
ECR
RPCR
G-BLR
DDCR
VMMR
SDi
SSi
(1)
(2)
(4)

FIG. 32

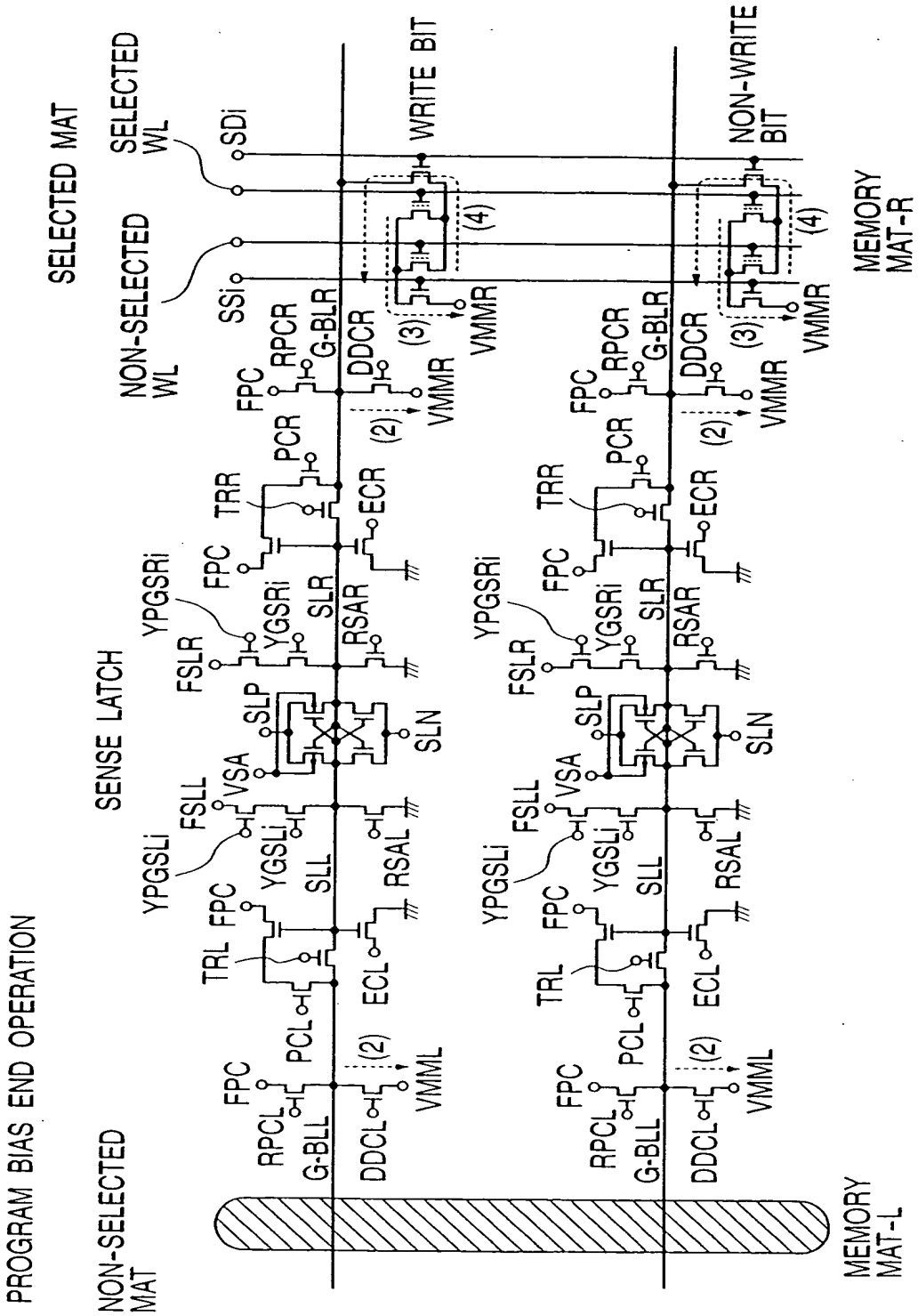

PROGRAM BIAS END OPERATION
NON-SELECTED MAT
SENSE LATCH
SELECTED MAT
NON-SELECTED WL
SELECTED WL
SDi
SSi
WRITE BIT
NON-WRITE BIT
MEMORY MAT-L
MEMORY MAT-R
YPGSLi
YPGSRi
FSLL
FSLR
VSA
SLP
SLN
YGSLi
YGSRi
SLL
SLR
RSAL
RSAR
TRL
TRR
PCL
PCR
ECL
ECR
FPC
RPCL
RPCR
G-BLL
G-BLR
DDCL
DDCR
VMML
VMMR
(2)
(3)
(4)

FIG. 33

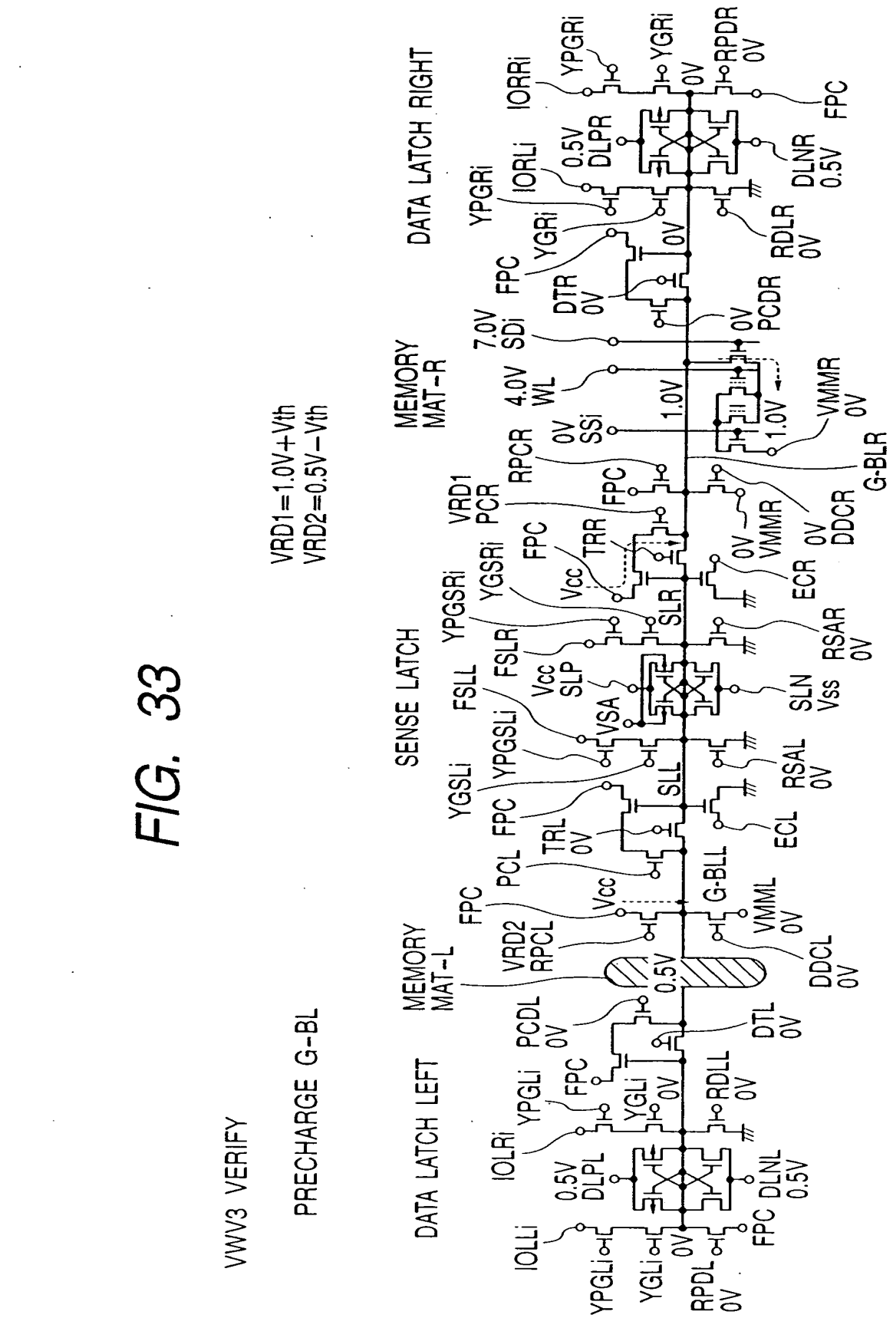

VWV3 VERIFY
PRECHARGE G-BL
VRD1=1.0V+Vth
VRD2=0.5V−Vth
DATA LATCH LEFT
DATA LATCH RIGHT
SENSE LATCH
MEMORY MAT-L
MEMORY MAT-R
IOLLi
IOLRi
YPGLi
YGLi
RPDL 0V
FPC
DLPL 0.5V
DLNL 0.5V
PCDL 0V
RDLL 0V
DTL 0V
PCL
VRD2
RPCL
TRL 0V
YGSLi
YPGSLi
FSLL
VSA
Vcc
SLP
SLL
G-BLL
VMML 0V
DDCL 0V
ECL
RSAL 0V
SLN
Vss
RSAR 0V
FSLR
YPGSRi
YGSRi
SLR
TRR
ECR
VMMR
VRD1
PCR
RPCR
SSi 0V
WL 4.0V
SDi 7.0V
DTR 0V
PCDR 0V
RDLR 0V
DLPR 0.5V
DLNR 0.5V
IORLi
IORRi
YPGRi
YGRi
RPDR 0V
1.0V
VMMR 0V
DDCR 0V
G-BLR

FIG. 34

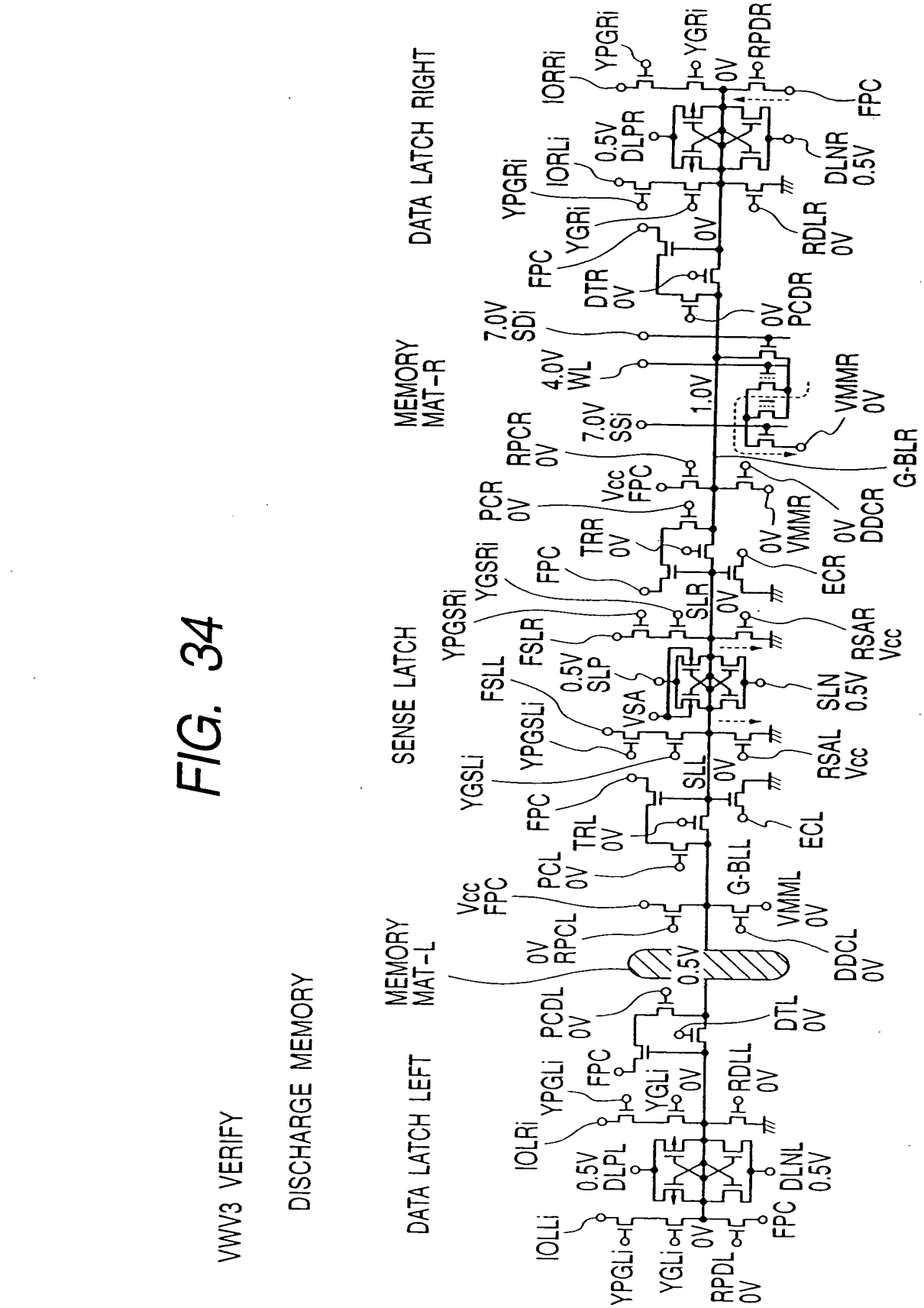

VWV3 VERIFY
DISCHARGE MEMORY
DATA LATCH LEFT
MEMORY MAT-L
SENSE LATCH
MEMORY MAT-R
DATA LATCH RIGHT
IOLLi
YPGLi
YGLi
RPDL 0V
0.5V DLPL
FPC DLNL 0.5V
IOLRi
YPGLi
YGLi 0V
FPC
PCDL 0V
RDLL 0V
DTL 0V
Vcc FPC
PCL 0V
RPCL 0V
TRL 0V
FPC
YGSLi
YPGSLi
FSLL
0.5V SLP
VSA
YPGSRi
YGSRi
FSLR
PCR 0V
RPCR 0V
TRR 0V
FPC
Vcc FPC
7.0V SSi
4.0V WL
7.0V SDi
FPC
YGRi
YPGRi
IORLi
0.5V DLPR
DTR 0V
RDLR 0V
DLNR 0.5V
IORRi
YPGRi
YGRi 0V
RPDR
FPC
SLL 0V
SLR 0V
RSAL Vcc
SLN 0.5V
RSAR Vcc
ECL
G-BLL
VMML 0V
DDCL 0V
0.5V
1.0V
PCDR 0V
VMMR 0V
ECR 0V
VMMR 0V
DDCR 0V
G-BLR

FIG. 35

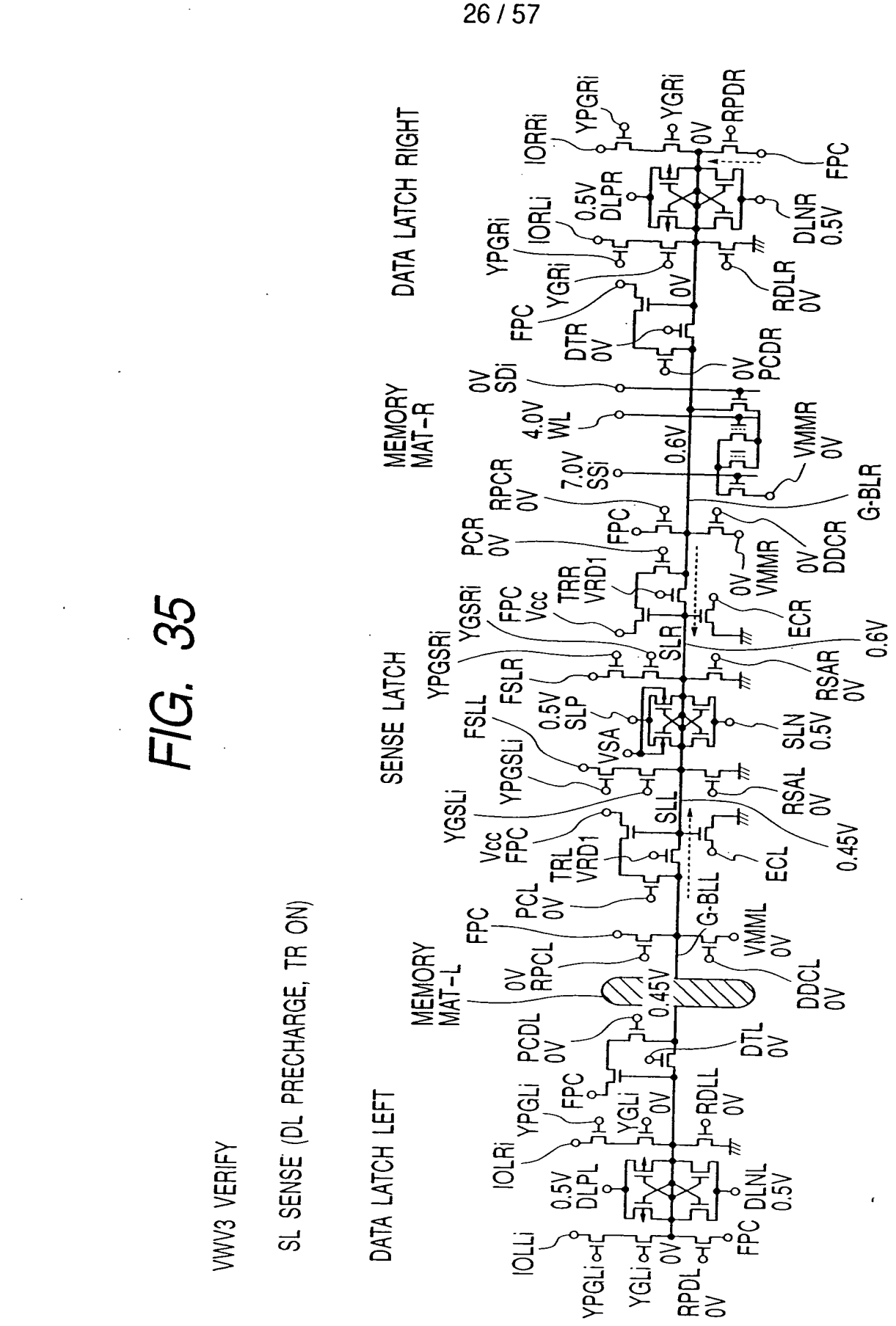

VWV3 VERIFY
SL SENSE (DL PRECHARGE, TR ON)
DATA LATCH LEFT
SENSE LATCH
DATA LATCH RIGHT
MEMORY MAT-L
MEMORY MAT-R
IOLRi
IOLLi
YPGLi
YGLi
RPDL
PCDL
RPCL
PCL
TRL
VRD1
YGSLi
YPGSLi
FSLL
FSLR
SLP
VSA
SLL
SLR
G-BLL
G-BLR
VMML
VMMR
DDCL
DDCR
ECL
ECR
RSAL
RSAR
SLN
DTL
DTR
RDLL
RDLR
DLPL
DLNL
DLPR
DLNR
PCR
RPCR
WL
SSi
SDi
PCDR
YGRi
YPGRi
IORRi
IORLi
RPDR
FPC
Vcc
0V
0.5V
0.45V
0.6V
4.0V
7.0V

FIG. 36

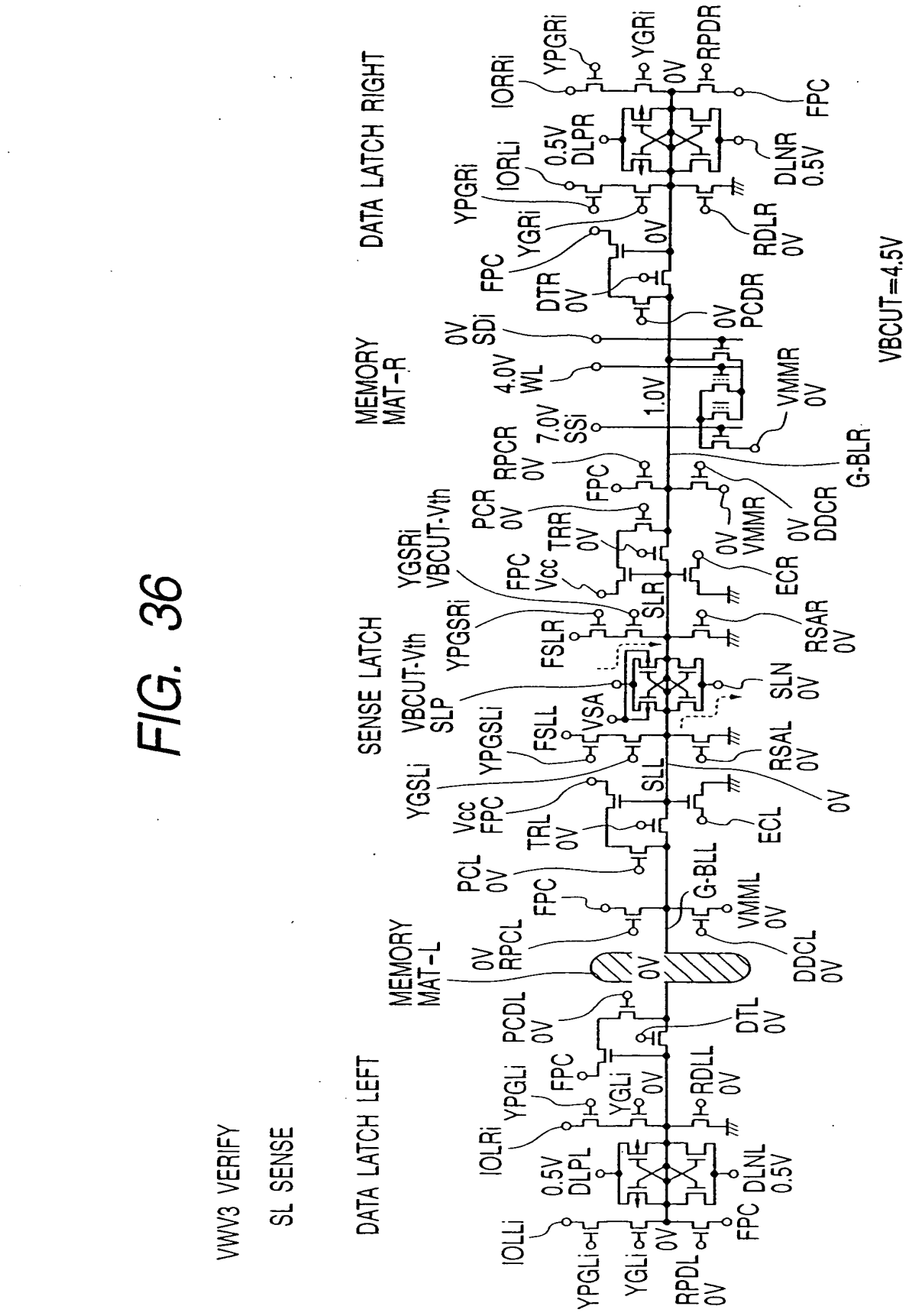

VWV3 VERIFY
SL SENSE
DATA LATCH LEFT
SENSE LATCH
DATA LATCH RIGHT
MEMORY MAT-L
MEMORY MAT-R
VBCUT=4.5V
IOLLi
IOLRi
IORLi
IORRi
YPGLi
YPGRi
YGLi
YGRi
RPDL
RPDR
PCDL
PCDR
DTL
DTR
RDLL
RDLR
DLPL
DLPR
DLNL
DLNR
0.5V
FPC
RPCL
RPCR
PCL
PCR
TRL
TRR
SLL
SLR
SLP
SLN
SDi
SSi
WL
4.0V
7.0V
1.0V
VMML
VMMR
DDCL
DDCR
ECL
ECR
RSAL
RSAR
G-BLL
G-BLR
YGSLi
YGSRi
YPGSLi
YPGSRi
FSLL
FSLR
VSA
VBCUT-Vth
Vcc
0V

FIG. 37

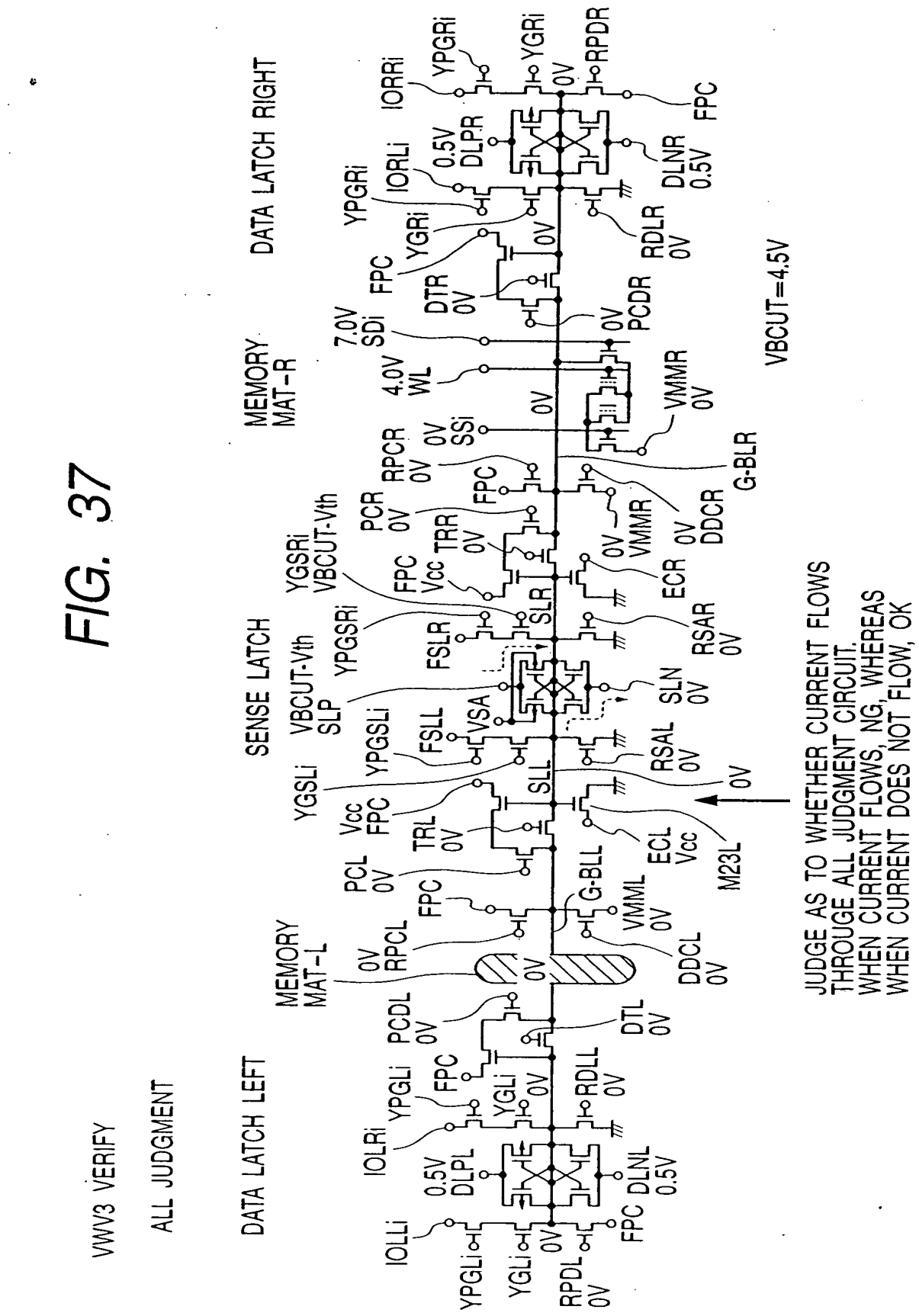

VWV3 VERIFY
ALL JUDGMENT
DATA LATCH LEFT
SENSE LATCH
DATA LATCH RIGHT
MEMORY MAT-L
MEMORY MAT-R
VBCUT=4.5V
JUDGE AS TO WHETHER CURRENT FLOWS THROUGE ALL JUDGMENT CIRCUIT.
WHEN CURRENT FLOWS, NG, WHEREAS
WHEN CURRENT DOES NOT FLOW, OK

FIG. 38

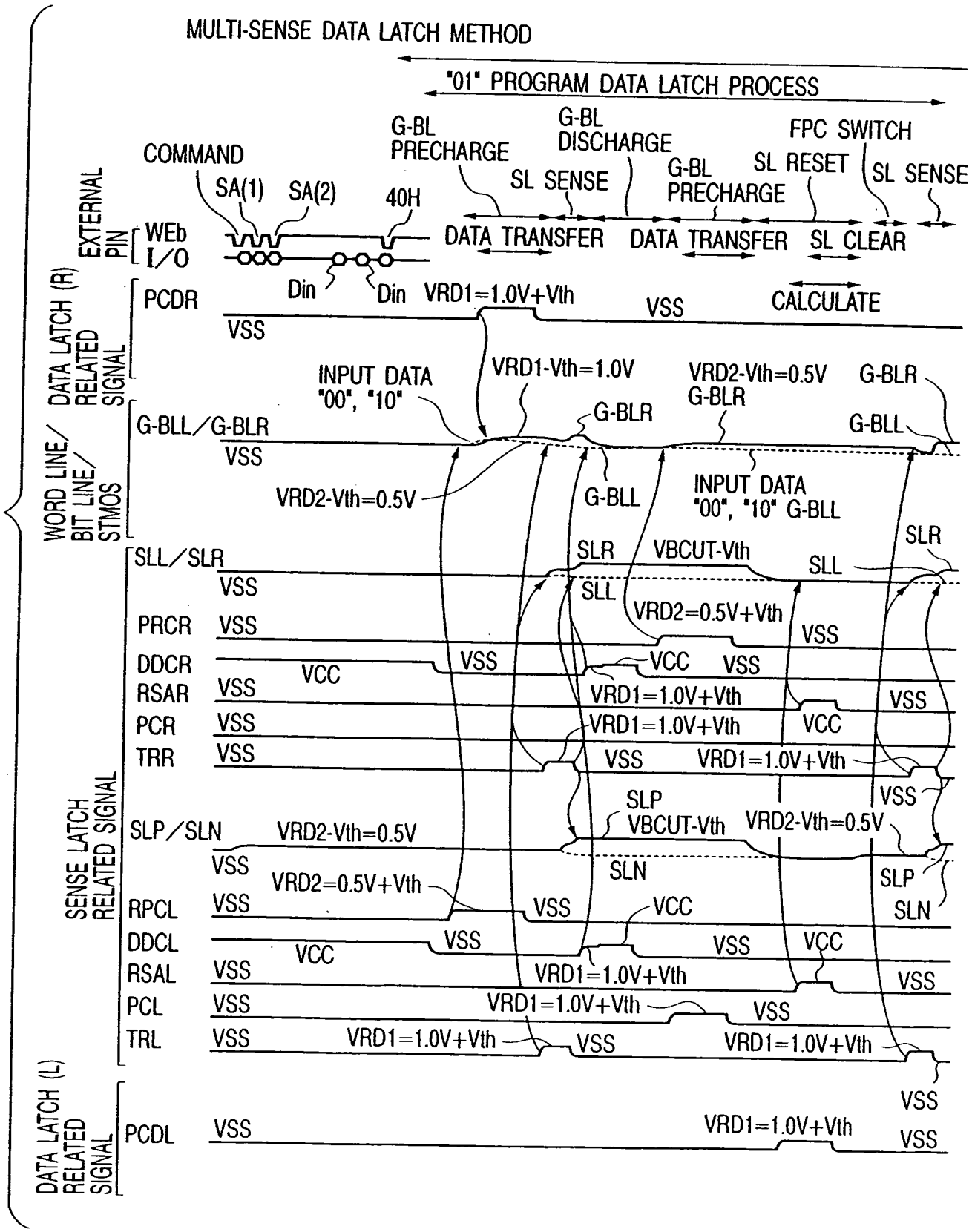

MULTI-SENSE DATA LATCH METHOD
"01" PROGRAM DATA LATCH PROCESS
G-BL PRECHARGE
G-BL DISCHARGE
FPC SWITCH
COMMAND
SA(1)
SA(2)
40H
SL SENSE
G-BL PRECHARGE
SL RESET
SL SENSE
EXTERNAL PIN
WEb
I/O
DATA TRANSFER
DATA TRANSFER
SL CLEAR
Din
Din
DATA LATCH (R) RELATED SIGNAL
PCDR
VRD1=1.0V+Vth
VSS
CALCULATE
VSS
INPUT DATA "00", "10"
VRD1-Vth=1.0V
VRD2-Vth=0.5V
G-BLR
G-BLR
G-BLR
G-BLL
WORD LINE/ BIT LINE/ STMOS
G-BLL/G-BLR
VSS
VRD2-Vth=0.5V
G-BLL
INPUT DATA "00", "10" G-BLL
SLR
SLR
VBCUT-Vth
SLL
SLL/SLR
VSS
SLL
VRD2=0.5V+Vth
PRCR
VSS
VSS
DDCR
VSS
VCC
VSS
VCC
RSAR
VSS
VRD1=1.0V+Vth
VSS
PCR
VSS
VRD1=1.0V+Vth
VCC
TRR
VSS
VSS
VRD1=1.0V+Vth
SLP
VBCUT-Vth
VRD2-Vth=0.5V
VSS
SLP/SLN
VRD2-Vth=0.5V
VSS
SLN
SLP
SLN
VRD2=0.5V+Vth
SENSE LATCH RELATED SIGNAL
RPCL
VSS
VSS
VCC
DDCL
VSS
VCC
VSS
VCC
RSAL
VSS
VRD1=1.0V+Vth
VSS
PCL
VSS
VRD1=1.0V+Vth
VSS
TRL
VSS
VRD1=1.0V+Vth
VSS
VRD1=1.0V+Vth
VSS
DATA LATCH (L) RELATED SIGNAL
PCDL
VSS
VRD1=1.0V+Vth
VSS

FIG. 39

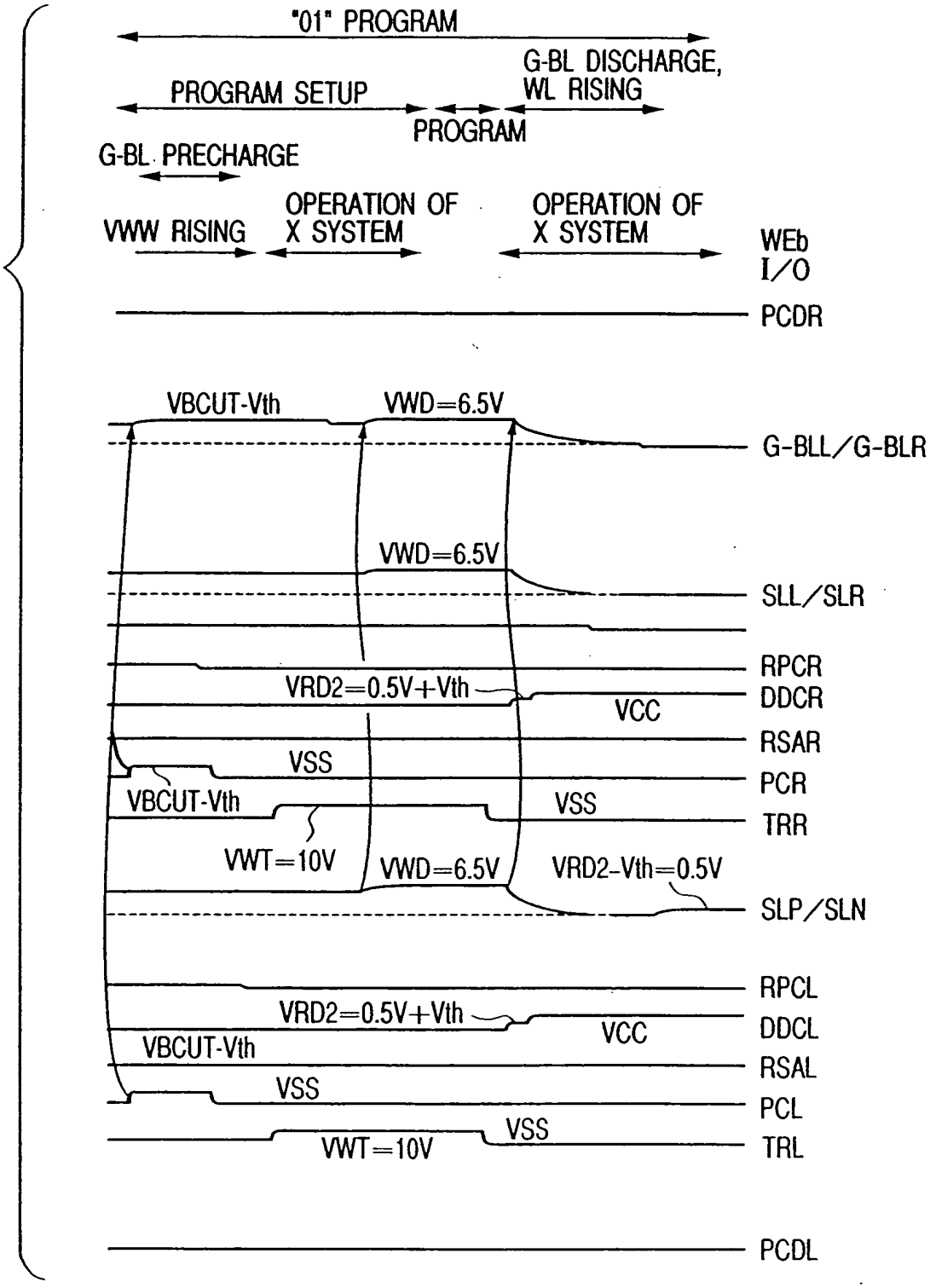

"01" PROGRAM
G-BL DISCHARGE, WL RISING
PROGRAM SETUP
PROGRAM
G-BL PRECHARGE
VWW RISING
OPERATION OF X SYSTEM
OPERATION OF X SYSTEM
WEb I/O
PCDR
VBCUT-Vth
VWD=6.5V
G-BLL/G-BLR
VWD=6.5V
SLL/SLR
RPCR
VRD2=0.5V+Vth
DDCR
VCC
RSAR
VSS
PCR
VBCUT-Vth
VSS
TRR
VWT=10V
VWD=6.5V
VRD2-Vth=0.5V
SLP/SLN
RPCL
VRD2=0.5V+Vth
DDCL
VCC
VBCUT-Vth
RSAL
VSS
PCL
VSS
VWT=10V
TRL
PCDL

FIG. 40

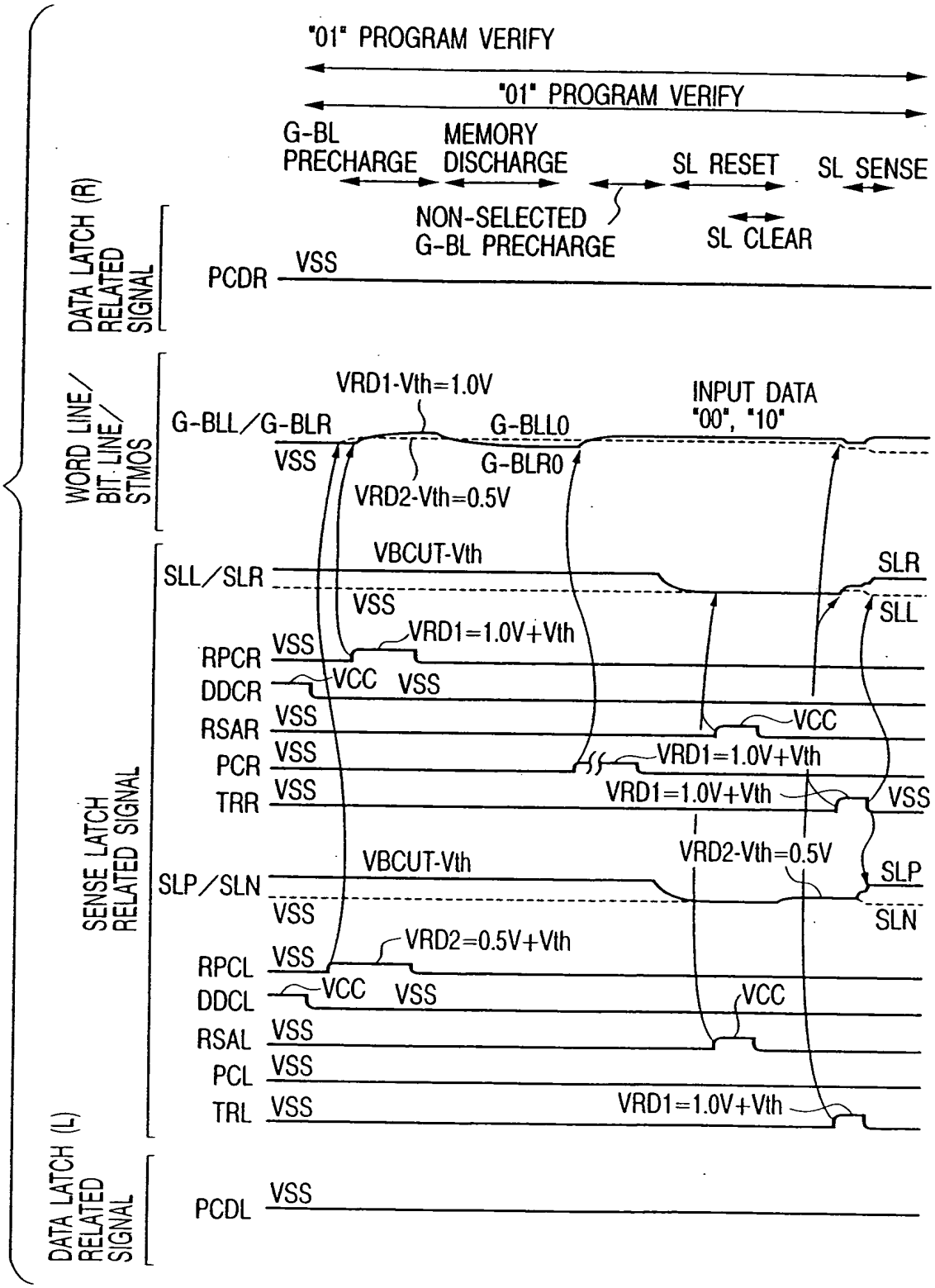

"01" PROGRAM VERIFY
"01" PROGRAM VERIFY
G-BL PRECHARGE
MEMORY DISCHARGE
SL RESET
SL SENSE
NON-SELECTED G-BL PRECHARGE
SL CLEAR
DATA LATCH (R) RELATED SIGNAL
PCDR
VSS
WORD LINE/ BIT LINE/ STMOS
G-BLL/G-BLR
VRD1-Vth=1.0V
G-BLL0
INPUT DATA "00", "10"
VSS
G-BLR0
VRD2-Vth=0.5V
SENSE LATCH RELATED SIGNAL
SLL/SLR
VBCUT-Vth
SLR
SLL
VSS
VRD1=1.0V+Vth
RPCR
DDCR
VCC
RSAR
PCR
TRR
SLP/SLN
VRD2-Vth=0.5V
SLP
SLN
VRD2=0.5V+Vth
RPCL
DDCL
RSAL
PCL
TRL
DATA LATCH (L) RELATED SIGNAL
PCDL

FIG. 41

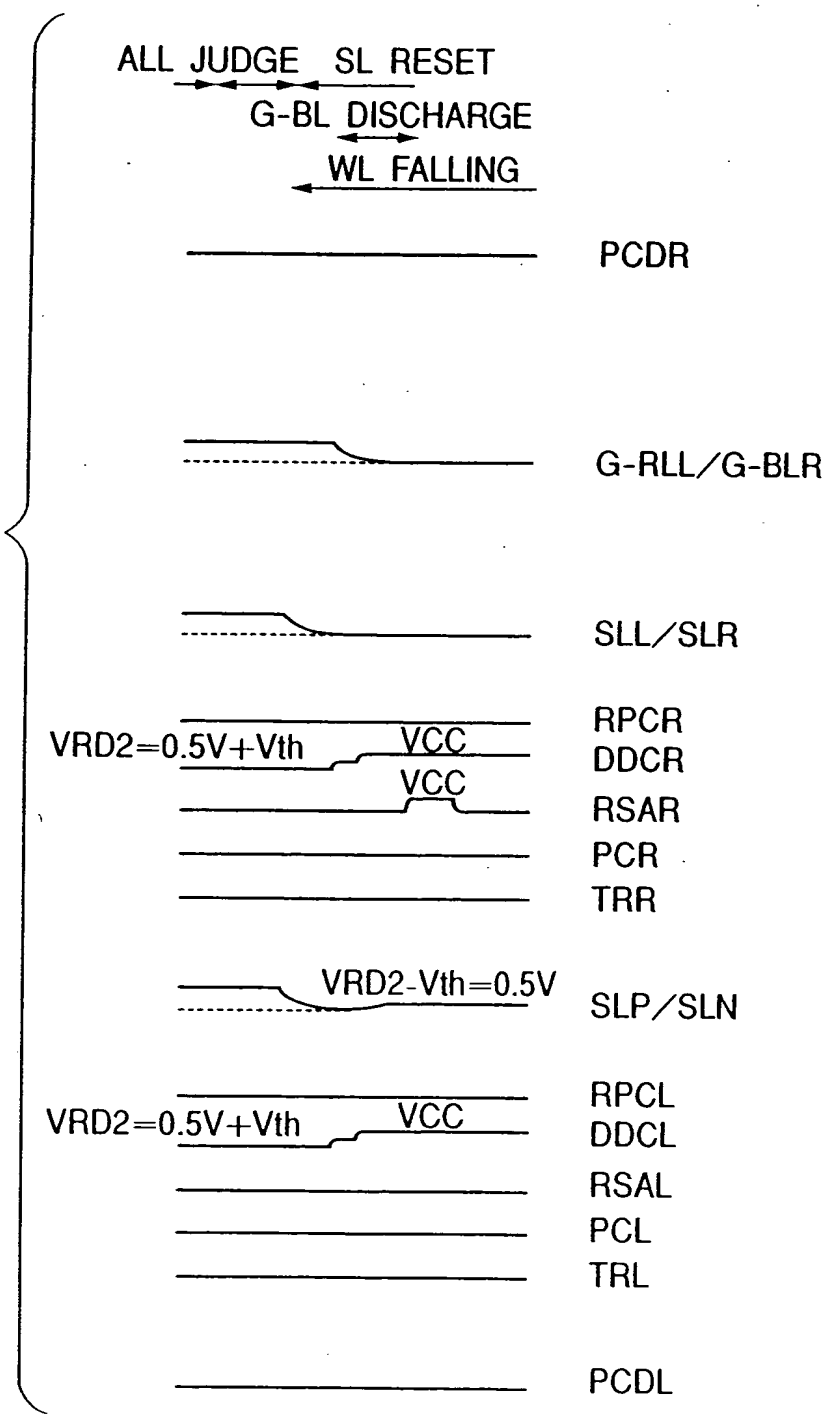
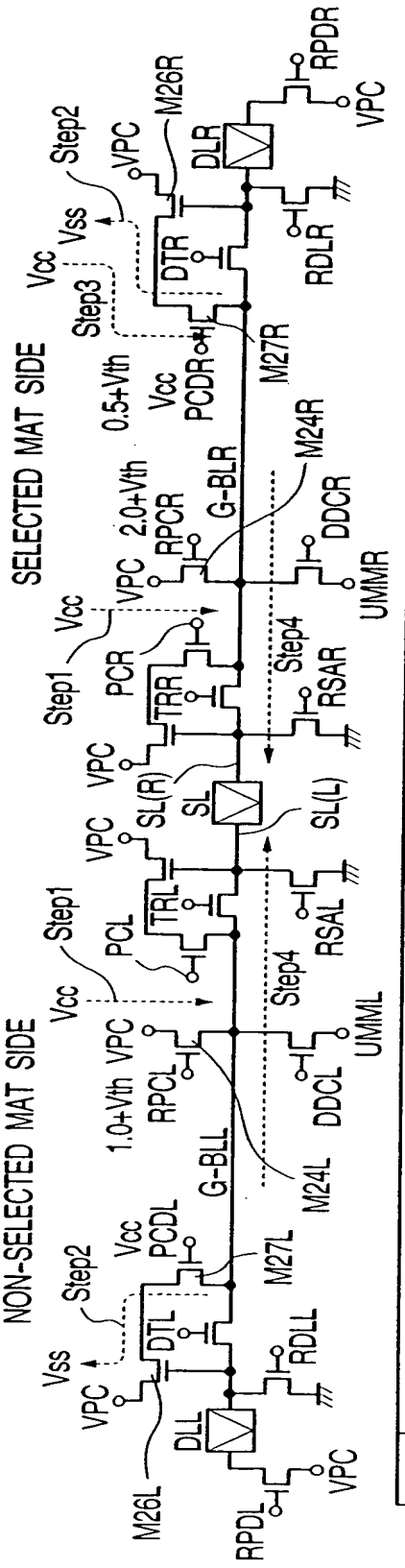

ALL JUDGE
SL RESET
G-BL DISCHARGE
WL FALLING
PCDR
G-RLL/G-BLR
SLL/SLR
RPCR
VRD2=0.5V+Vth
VCC
DDCR
VCC
RSAR
PCR
TRR
VRD2-Vth=0.5V
SLP/SLN
RPCL
VRD2=0.5V+Vth
VCC
DDCL
RSAL
PCL
TRL
PCDL

FIG. 42



Step	CONTENT		Step 1 PRECHARGE ALL G-BLR/L							Step 2 CALCULATE (DLR, G-BLR) CALCULATE (DLL, G-BLL)							Step 3 DATA TRANSFER DLR → G-BLR							Step 4 SL SENSE						
			DLL	G-BLL	SL(L)	SL(R)	G-BLR	DLR	DLL	G-BLL	SL(L)	SL(R)	G-BLR	DLR	DLL	G-BLL	SL(L)	SL(R)	G-BLR	DLR	DLL	G-BLL	SL(L)	SL(R)	G-BLR	DLR	DLL	G-BLL	SL(L)	SL(R)	G-BLR	DLR
PROGRAM DATA LATCH	01	0	1.0				2.0	1	0	1.0			0.0	1	0	1.0			0.5	1	0	1	1	0	0	1	1	0	1	1	1
	00	0	1.0				2.0	0	0	1.0			2.0	0	0	1.0			2.0	0	0	0	0	1	1	1	0	1	1	0	0
	10	1	1.0				2.0	0	1	0.0			2.0	0	1	0.0			2.0	0	1	0	0	1	1	0	1	0	0	1	1
	11	1	1.0				2.0	1	1	0.0			0.0	1	1	0.0			0.5	1	1	0	0	1	1	1	1	0	0	1	1

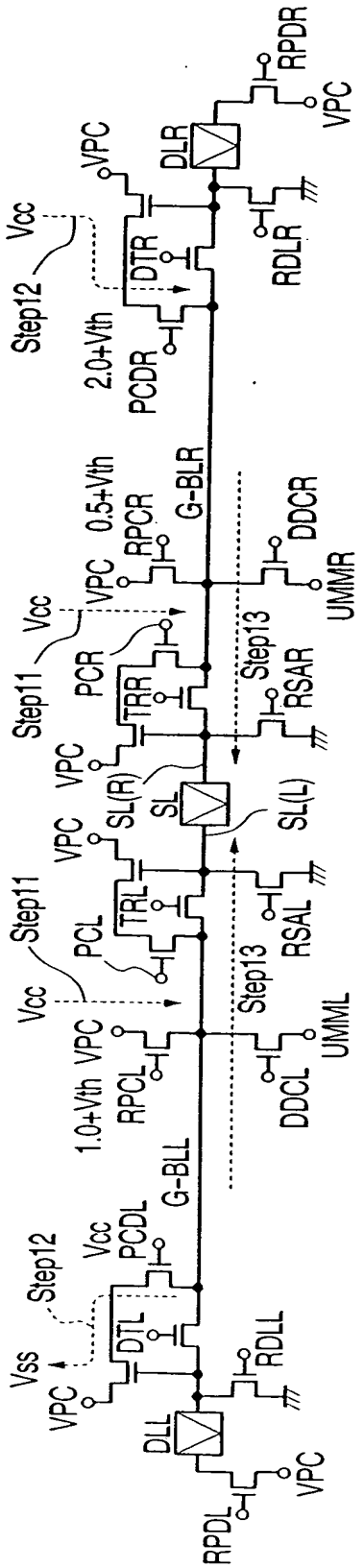
(a) LATCH DATA TO BIT LINE SIDE OF DATA LATCH CIRCUIT ON NON-SELECTED MAT SIDE

(b) (c) LATCH DATA TO BIT LINE SIDE OF DATA LATCH CIRCUIT ON SELECTED MAT SIDE IN CASE OF MULTI-POWER SUPPLY METHOD

(d) (e) (f) PROCESSED RESULT

'01' PROGRAM DATA LATCH PROCESS OPERATION (MULTI-POWER SUPPLY METHOD)

FIG. 43

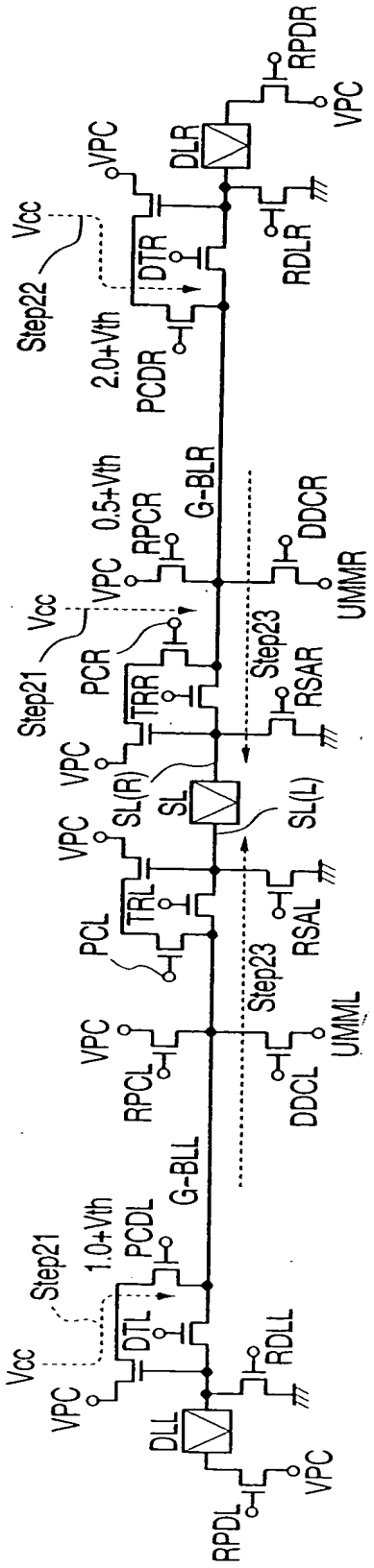


Step		PROGRAM DATA LATCH										
		Step 11				Step 12				Step 13		
CONTENT		PRECHARGE ALL G-BLR/L				CALCULATE (DLL, G-BLL) DATA TRANSFER DLR → G-BLR				SL SENSE		
		DLL	G-BLL	SL(L)	SL(R)	G-BLR	DLR	DLL	G-BLL	SL(L)	SL(R)	G-BLR	DLR
01	0	1.0			0.5	1.0	0	1				2.0	1
00	0	1.0			0.5	1.0	0	0				0.5	0
10	1	1.0			0.5	0.0	1	0				0.5	0
11	1	1.0			0.5	0.0	1	1				2.0	1

Step		DLL	G-BLL	SL(L)	SL(R)	G-BLR	DLR	DLL	G-BLL	SL(L)	SL(R)	G-BLR	DLR
01	0	0	1	0	0	1	1	0	0	0	1	1	1
00	0	0	1	0	0	1	1	0	1	1	0	0	0
10	1	1	1	0	0	1	1	1	0	0	1	1	0
11	1	1	1	0	0	1	1	1	0	1	1	1	1

'00' PROGRAM DATA LATCH PROCESS OPERATION
(MULTI-POWER SUPPLY METHOD)

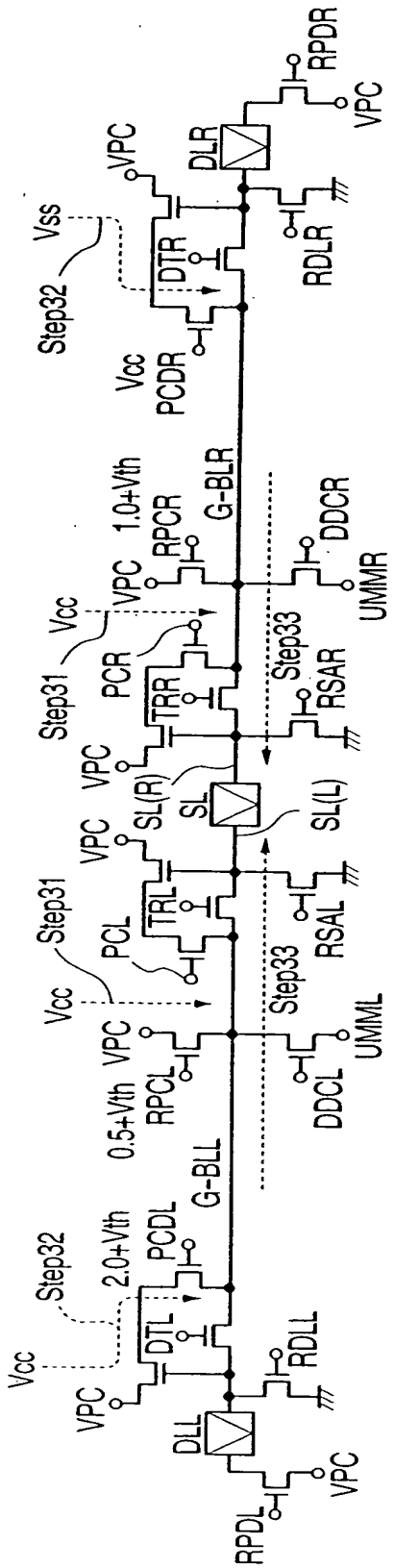
FIG. 44



Step	CONTENT																
	PROGRAM DATA LATCH																
	Step 21					Step 22					Step 23						
	PRECHARGE ALL G-BLR DATA TRANSFER DLL → G-BLL					DATA TRANSFER DLR → G-BLR					SL SENSE						
	DLL	G-BLL	SL(L)	SL(R)	G-BLR	DLR	DLL	G-BLL	SL(L)	SL(R)	G-BLR	DLR	DLL	G-BLL	SL(L)	SL(R)	G-BLR	DLR
01	0	0.0			0.5	1	0	0.0			2.0	1	0	0	0	1	1	1
00	0	0.0			0.5	0	0	0.0			0.5	0	0	0	0	1	1	0
10	1	1.0			0.5	0	1	1.0			0.5	0	1	1	1	0	0	0
11	1	1.0			0.5	1	1	1.0			2.0	1	1	0	0	1	1	1

'10' PROGRAM DATA LATCH PROCESS OPERATION
(MULTI-POWER SUPPLY METHOD)

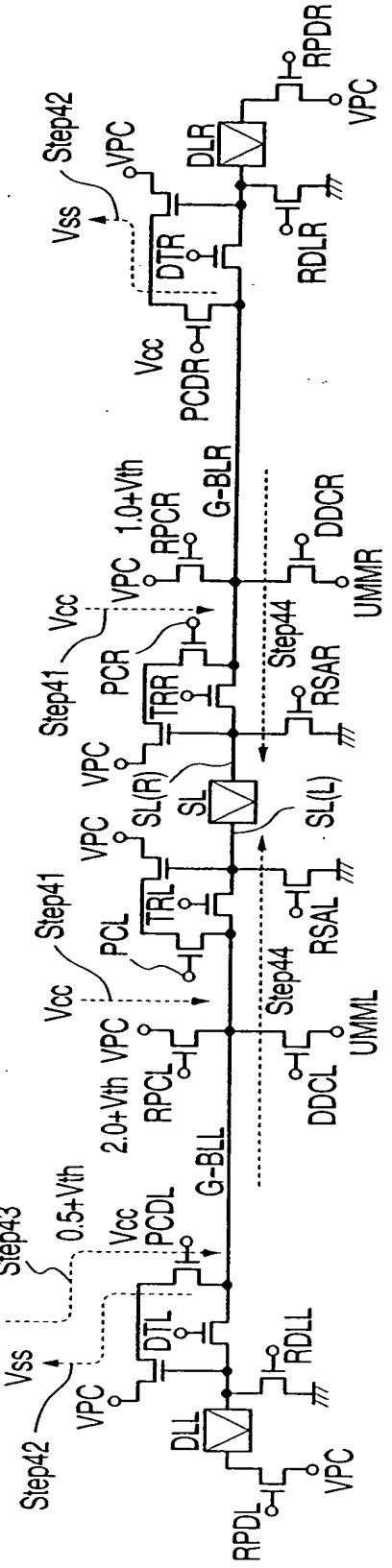
FIG. 45



Step		ERRATIC DATA LATCH													
		Step 31				Step 32				Step 33					
CONTENT		PRECHARGE ALL G-BLR/L				CALCULATE (DLR, G-BLR) DATA TRANSFER DLL → G-BLL				SL SENSE					
		DLL	G-BLL	SL(L)	SL(R)	G-BLR	DLR	DLL	G-BLL	SL(L)	SL(R)	G-BLR	DLR	G-BLL	SL(L)	SL(R)	G-BLR	DLR
01	0	0.5	0.5			1.0	1	0	0.5			0.0	1	1	1	0	0	1
00	0	0.5	0.5			1.0	0	0	0.5			1.0	0	0	0	1	1	0
10	1	0.5	0.5			1.0	0	1	2.0			1.0	0	1	1	0	0	0
11	1	0.5	0.5			1.0	1	1	2.0			0.0	1	1	1	0	0	1

'00' ERRATIC DETECTION DATA LATCH PROCESS
(MULTI-POWER SUPPLY METHOD)

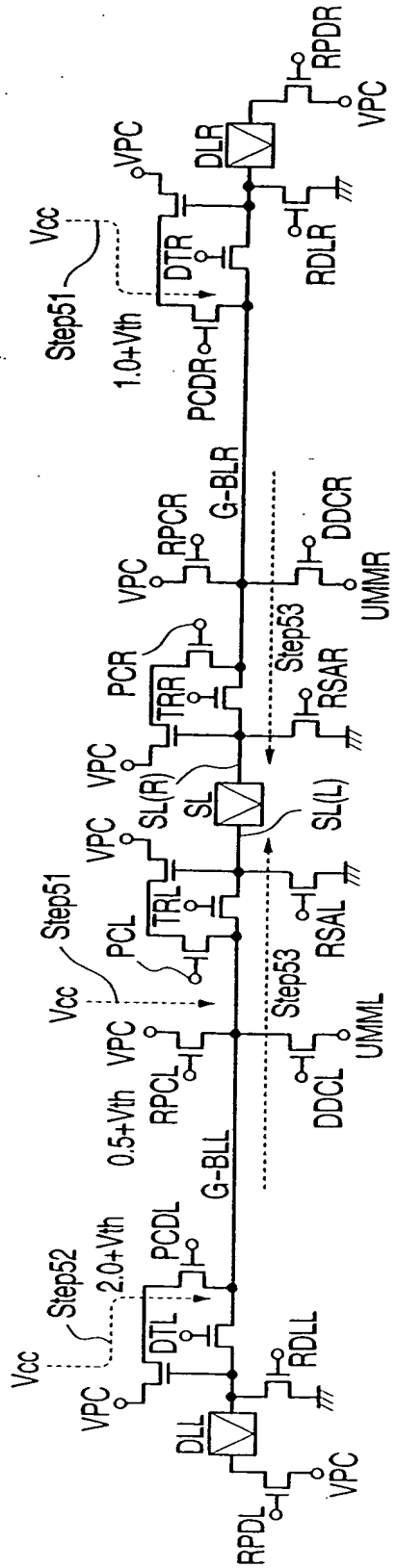
FIG. 46



Step		ERRATIC DATA LATCH																					
		Step 41				Step 42				Step 43				Step 44									
CONTENT		PRECHARGE ALL G-BLR/L				CALCULATE (DLR, G-BLR) CALCULATE (DLL, G-BLD)				DATA TRANSFER DLL → G-BLL				SL SENSE									
	DLL	G-BLL	SL(L)	SL(R)	G-BLR	DLL	G-BLL	SL(L)	SL(R)	DLR	G-BLR	DLL	G-BLL	SL(L)	SL(R)	G-BLR	DLR	DLL	G-BLL	SL(L)	SL(R)	G-BLR	DLR
01	0	2.0			1.0	0	2.0			1	0.0	0	2.0			0.0	1	0	1	1	0	0	1
00	0	2.0			1.0	0	2.0			0	1.0	0	2.0			1.0	0	0	1	1	0	0	0
10	1	2.0			1.0	1	10.0			0	1.0	1	0.5			1.0	0	1	0	0	1	1	0
11	1	2.0			1.0	1	10.0			1	0.0	1	0.5			0.0	1	1	1	1	0	0	1

'10' ERRATIC DETECTION DATA LATCH PROCESS OPERATION
(MULTI-POWER SUPPLY METHOD)

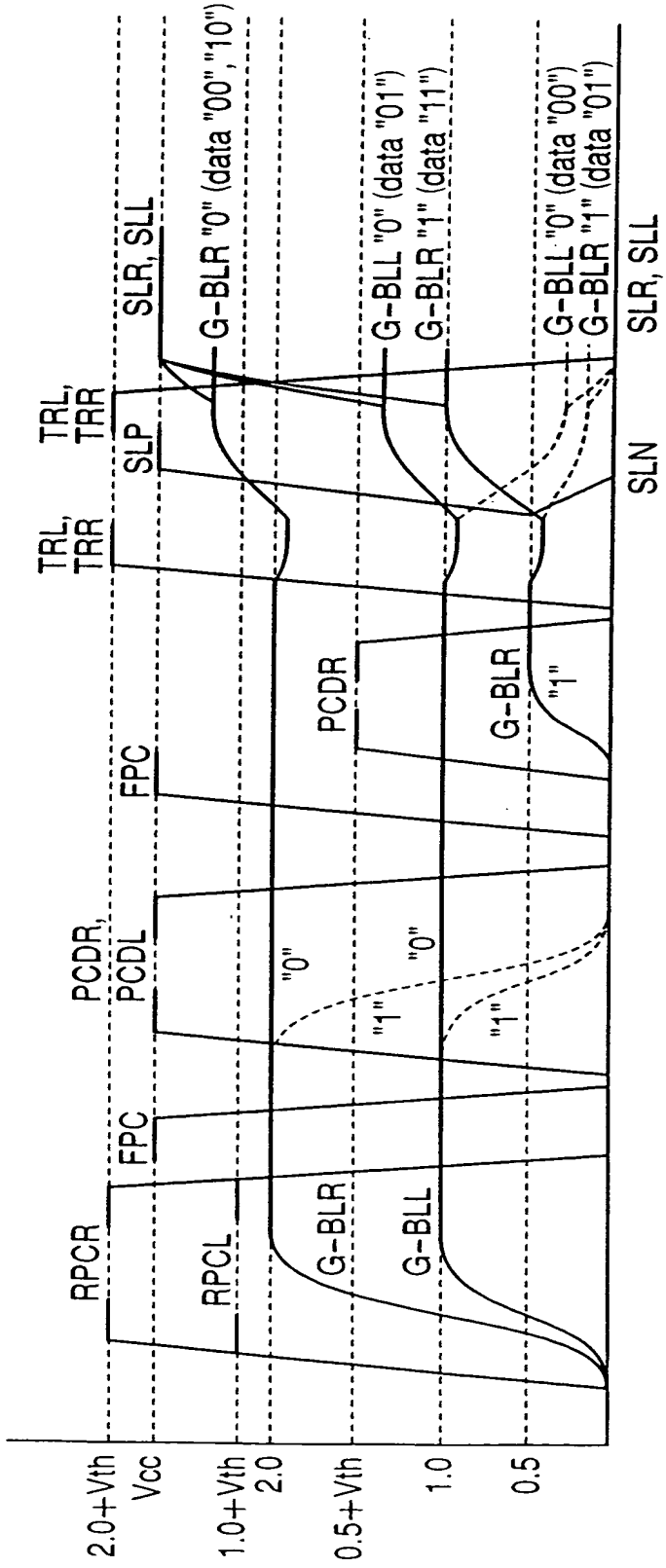
FIG. 47



Step		ERRATIC DATA LATCH																			
		Step 51				Step 52				Step 53				Step 54							
CONTENT		PRECHARGE ALL G-BLR/L				CALCULATE (DLL, G-BLD)				DATA TRANSFER DLL → G-BLL				SL SENSE							
		DLL	G-BLL	SL(L)	SL(R)	G-BLR	DLR	DLL	G-BLL	SL(L)	SL(R)	G-BLR	DLR	DLL	G-BLL	SL(L)	SL(R)	G-BLR	DLR	DLL	G-BLL	SL(L)	SL(R)	G-BLR	DLR
01	0	2.0	2.0			1.0	1	0	2.0			1.0	1	0	1	1	0	0	1						
00	0	2.0	2.0			0.0	0	0	2.0			0.0	0	0	1	1	0	0	0						
10	1	2.0	2.0			0.0	0	1	0.0			0.0	0	1	1	1	0	0	0						
11	1	2.0	2.0			1.0	1	1	0.0			1.0	1	1	0	0	1	1	1						

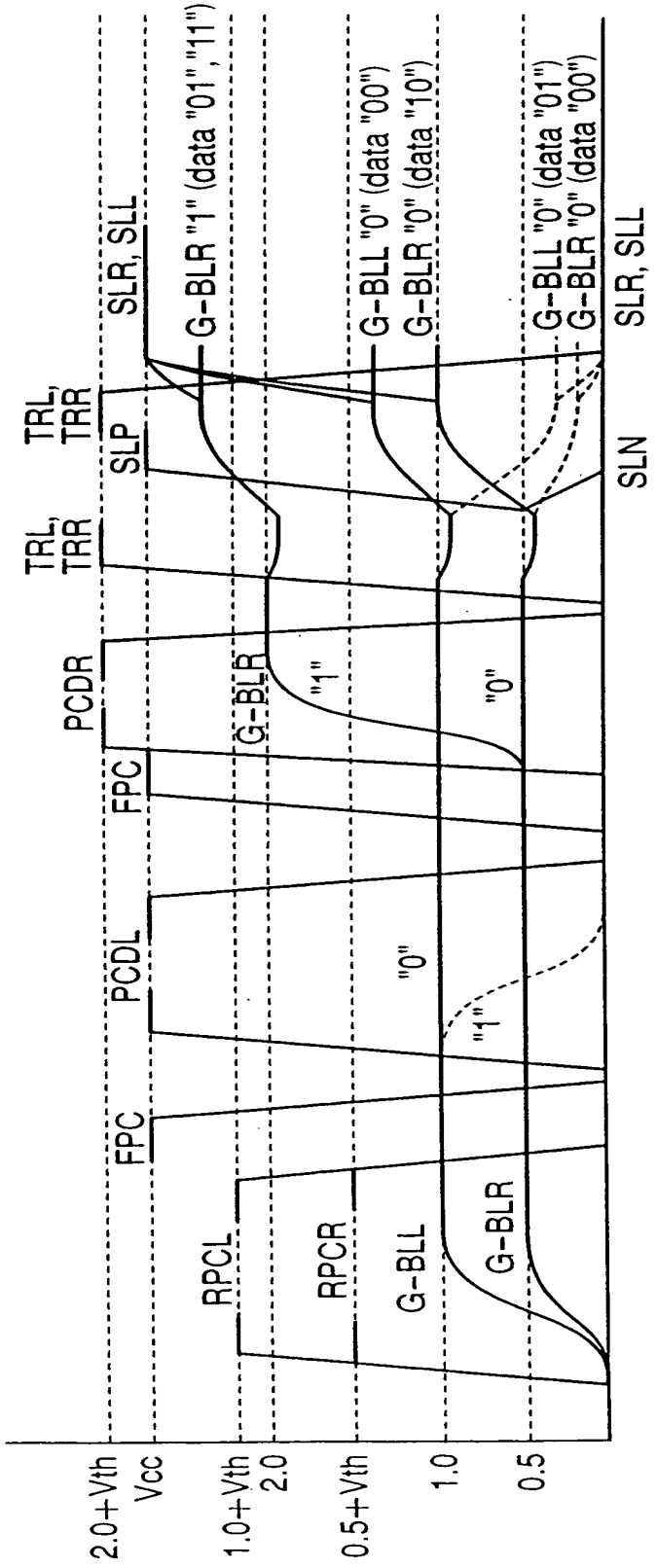
'11' DISTURB DETECTION DATA LATCH PROCESS OPERATION
(MULTI-POWER SUPPLY METHOD)

FIG. 48



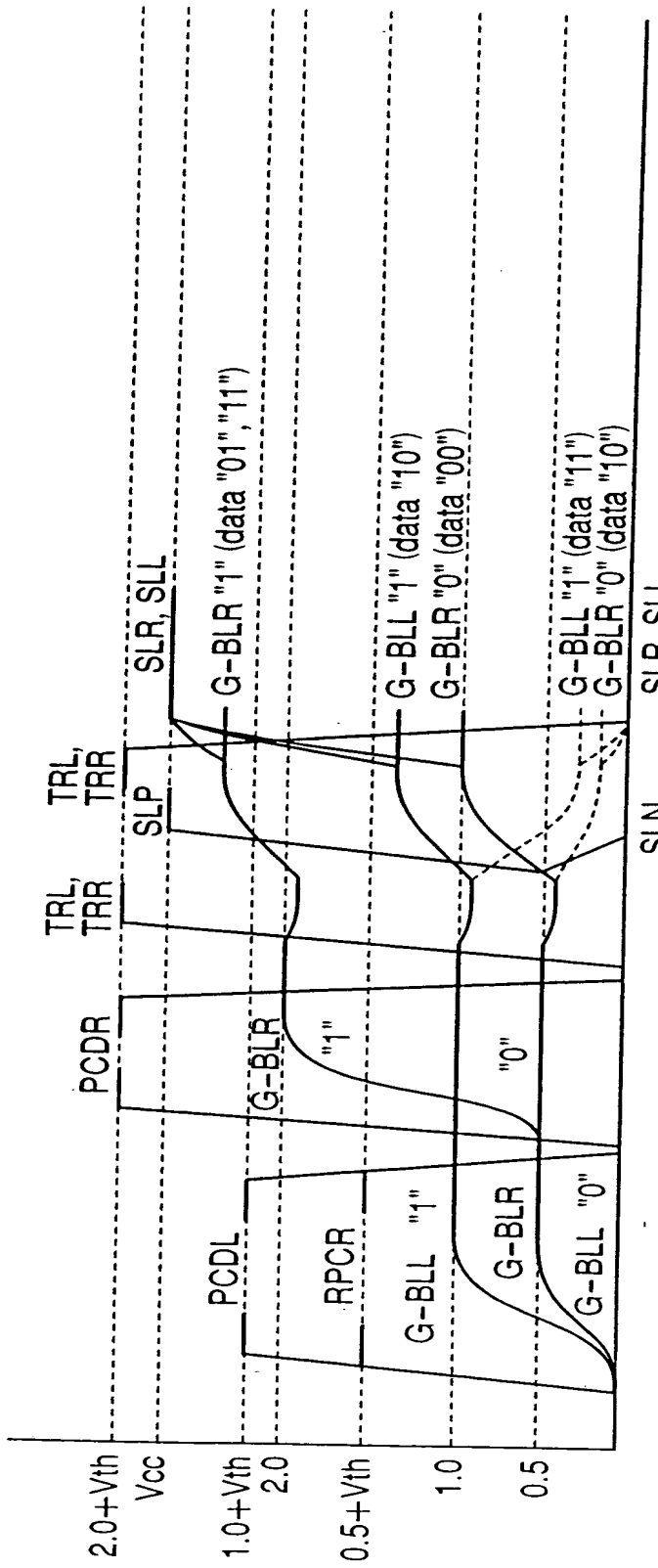
"01" PROGRAM DATA LATCH PROCESS WAVEFORM
(R-SIDE SELECTED IN MULTI-POWER SUPPLY METHOD)

FIG. 49



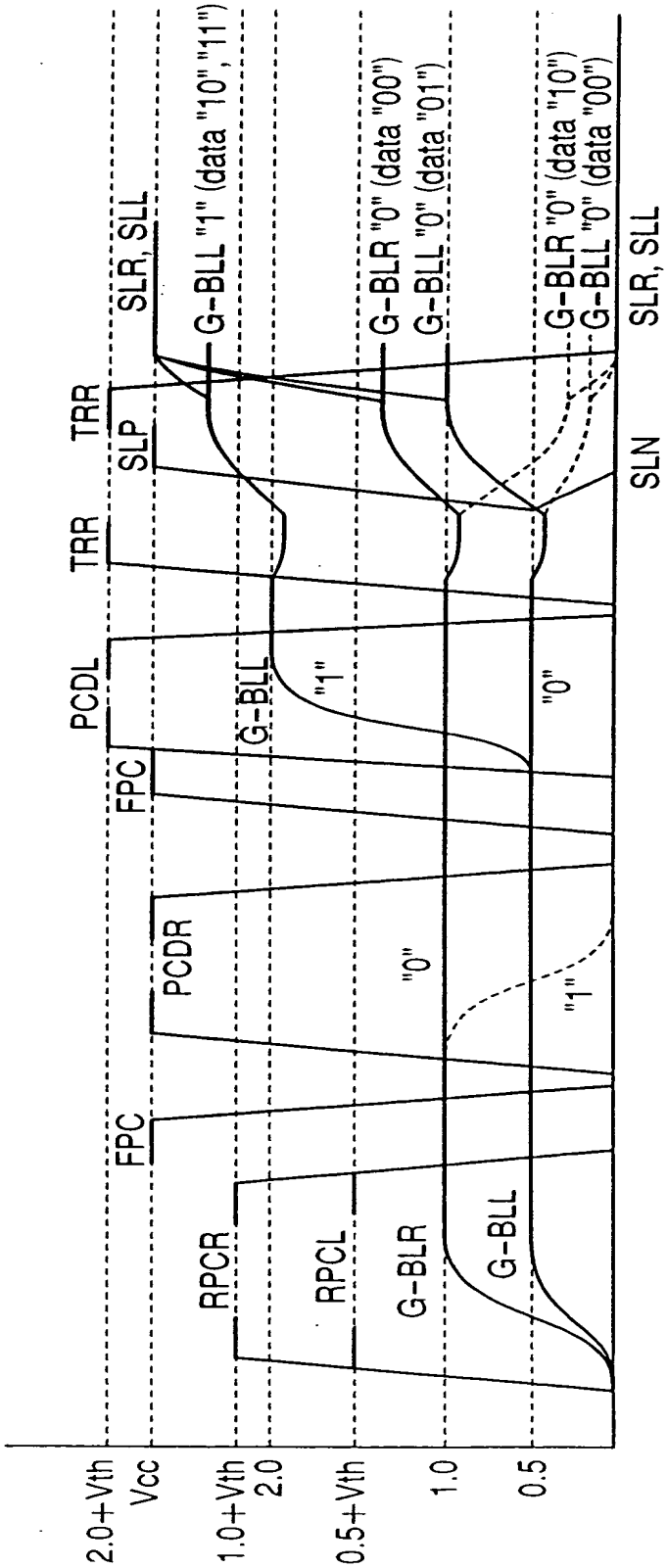
"00" PROGRAM DATA LATCH PROCESS WAVEFORM
(R-SIDE SELECTED IN MULTI-POWER SUPPLY METHOD)

FIG. 50



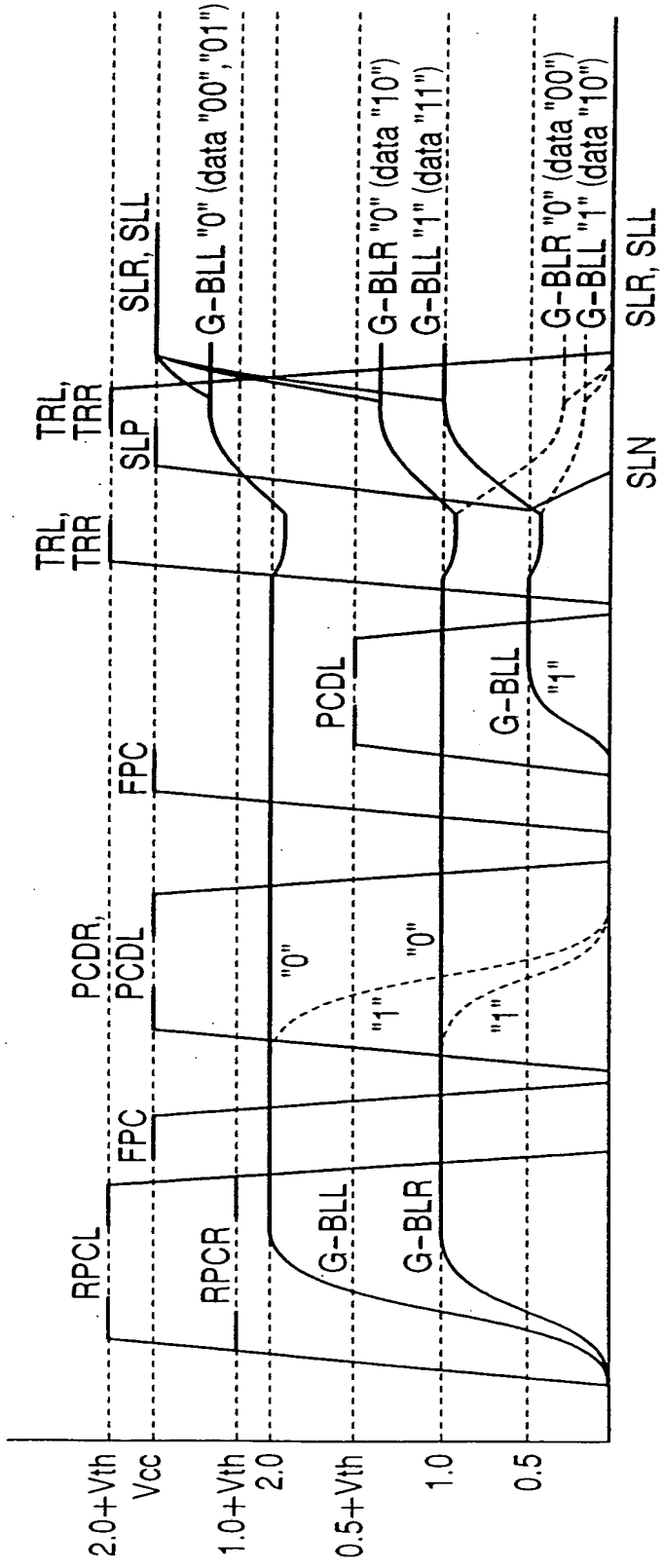
"10" PROGRAM DATA LATCH PROCESS WAVEFORM
(R-SIDE SELECTED IN MULTI-POWER SUPPLY METHOD)

FIG. 51



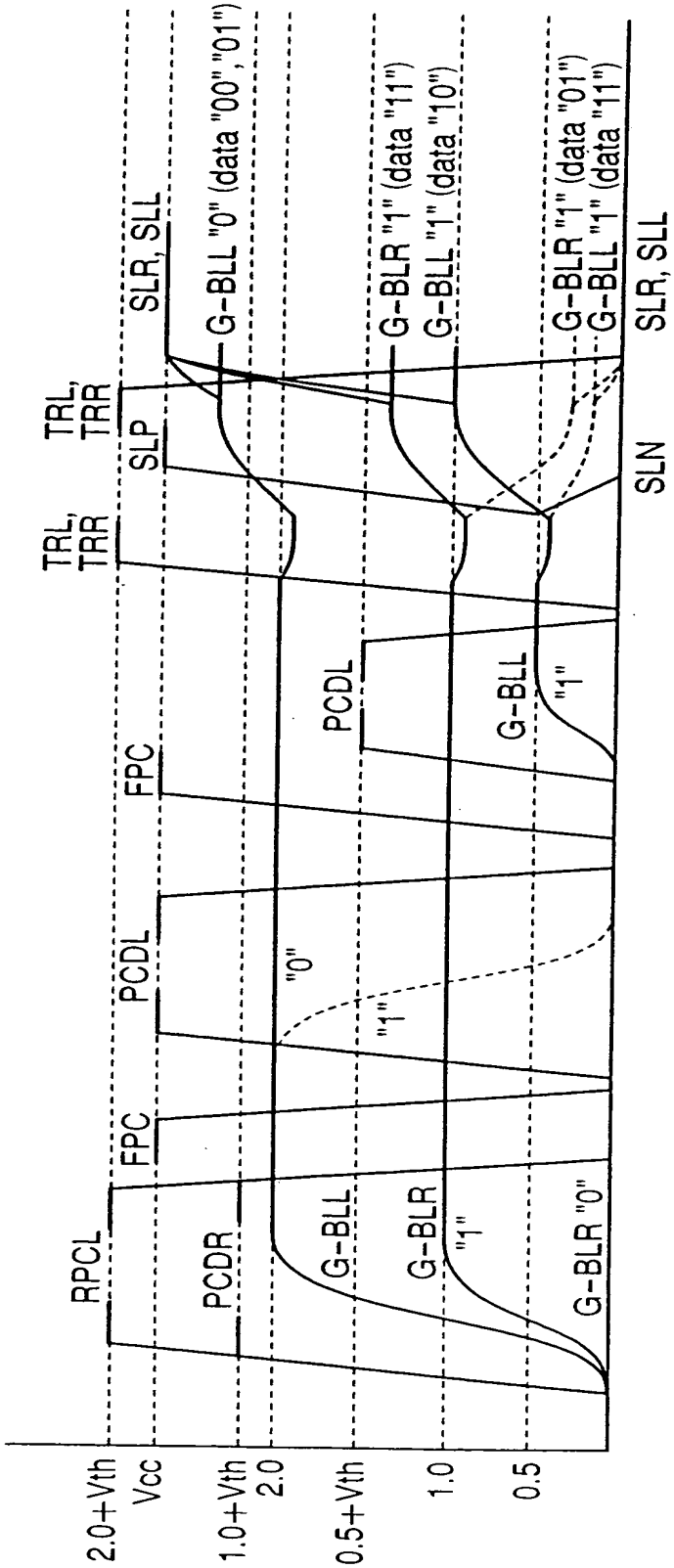
"00" ERRATIC DETECTION DATA LATCH PROCESS WAVEFORM
(R-SIDE SELECTED IN MULTI-POWER SUPPLY METHOD)

FIG. 52



"10" ERRATIC DETECTION DATA LATCH PROCESS WAVEFORM
(R-SIDE SELECTED IN MULTI-POWER SUPPLY METHOD)

FIG. 53



"11" DISTURB DETECTION DATA LATCH PROCESS WAVEFORM
(R-SIDE SELECTED IN MULTI-POWER SUPPLY METHOD)

FIG. 54

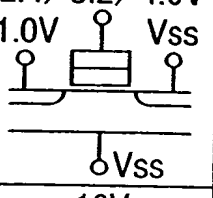
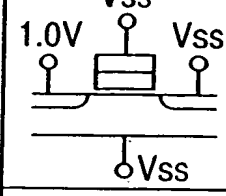
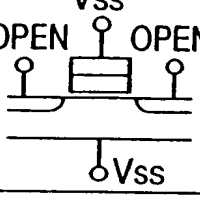
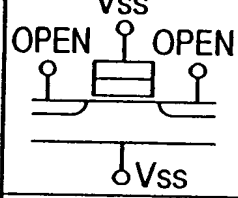
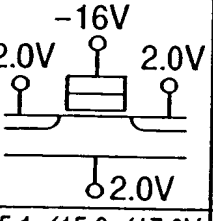
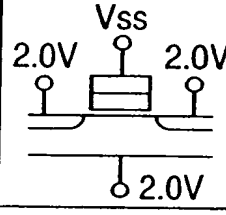
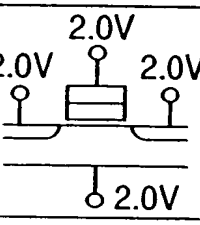
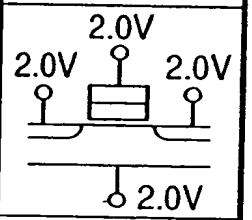
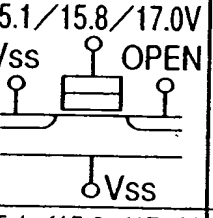
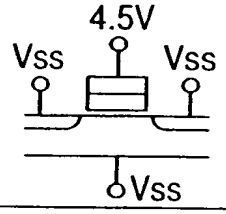
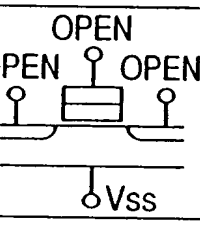
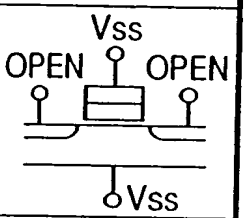
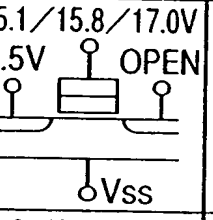
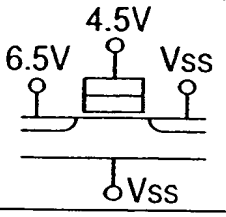
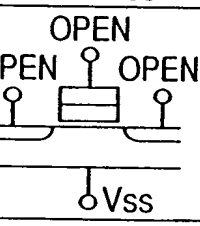
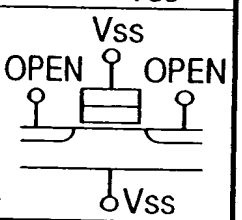
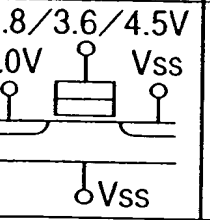
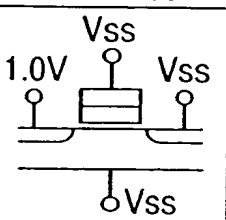
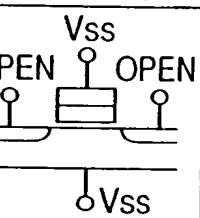
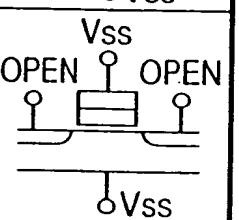
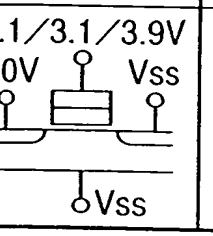
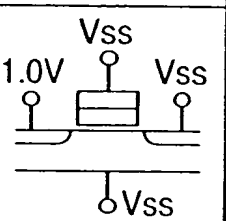
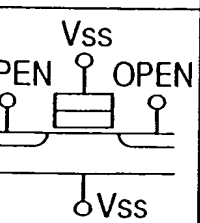
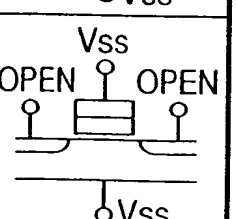
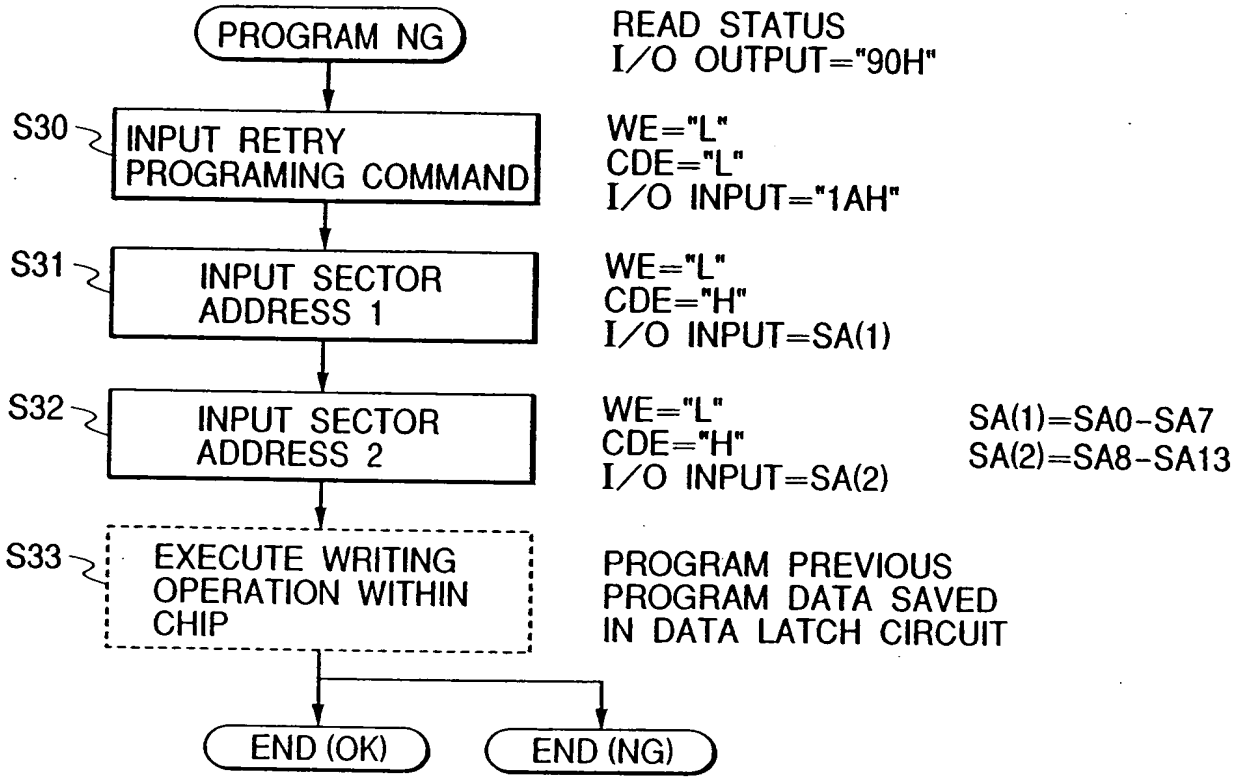
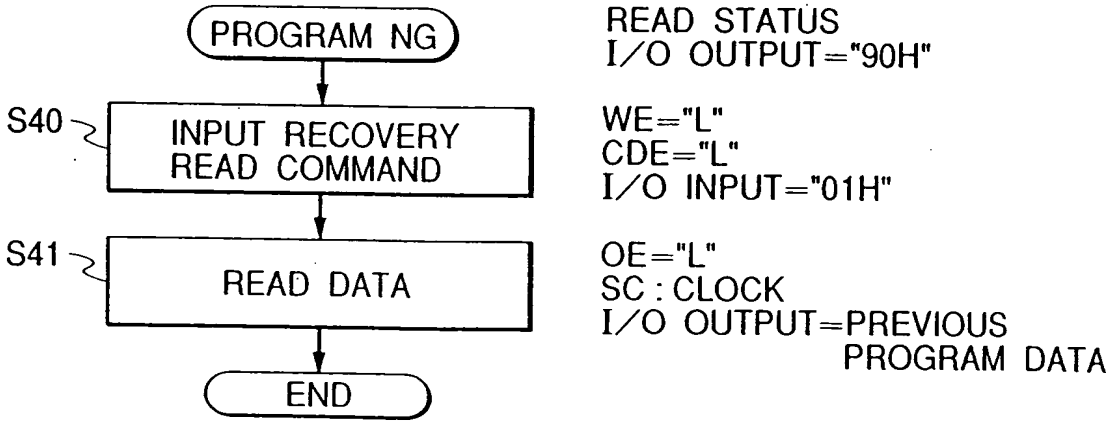
		SELECTED BLOCK		NON-SELECTED BLOCK	
		SELECTED WORD	NON-SELECTED WORD	SELECTED WORD	NON-SELECTED WORD
READ					
ERASE					
PROGRAM	PROGRAM DATA				
	NON-PROGRAM DATA				
	VERIFY				
	ERRATIC DETECT DISTURB DETECT				

FIG. 55



RETRY PROGRAM FLOW

FIG. 56



RECOVERY READ FLOW

FIG. 57

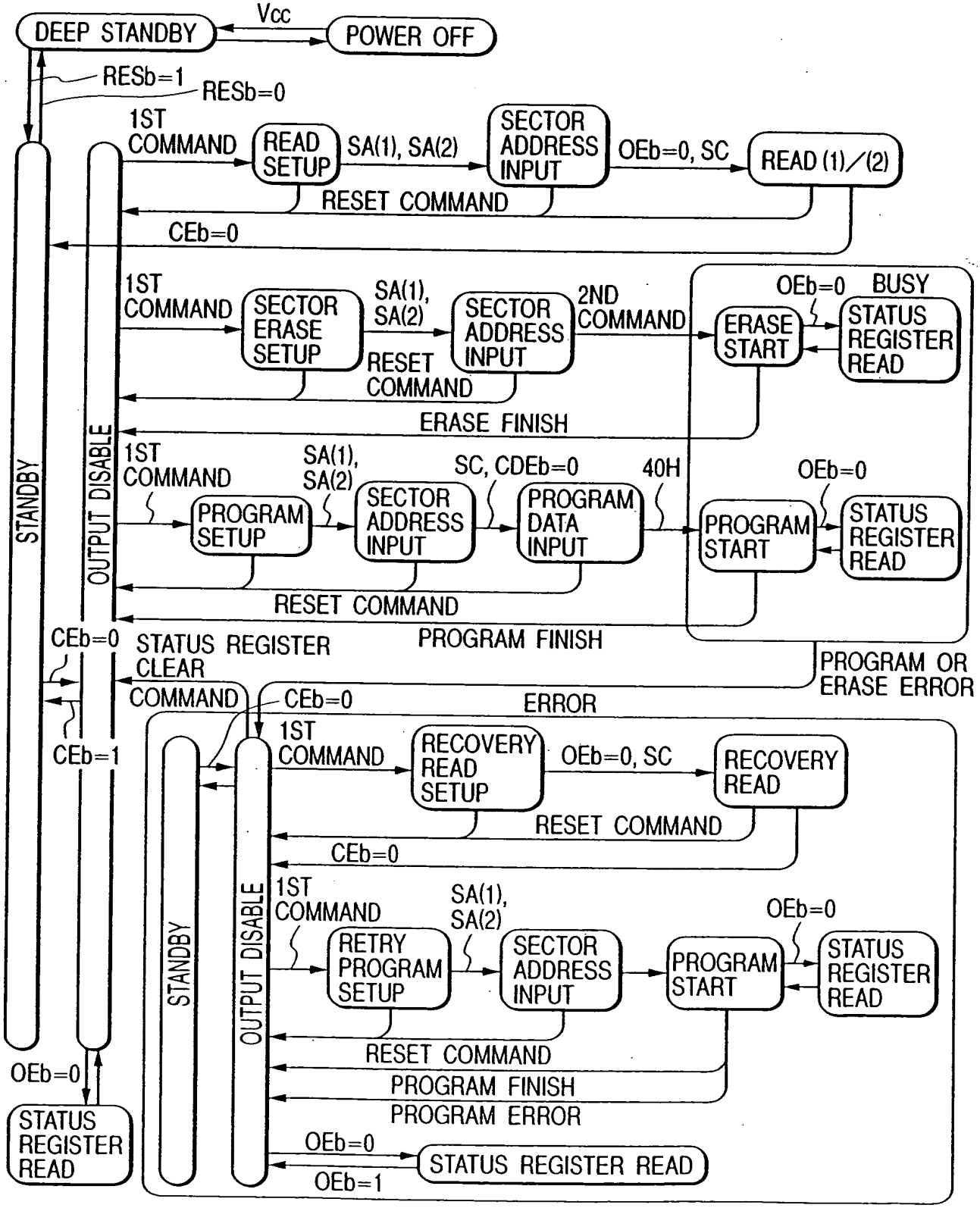

DEEP STANDBY
Vcc
POWER OFF
RESb=1
RESb=0
1ST COMMAND
READ SETUP
SA(1), SA(2)
SECTOR ADDRESS INPUT
OEb=0, SC
READ (1)/(2)
RESET COMMAND
CEb=0
1ST COMMAND
SECTOR ERASE SETUP
SA(1), SA(2)
SECTOR ADDRESS INPUT
2ND COMMAND
OEb=0
BUSY
ERASE START
STATUS REGISTER READ
RESET COMMAND
ERASE FINISH
1ST COMMAND
PROGRAM SETUP
SA(1), SA(2)
SECTOR ADDRESS INPUT
SC, CDEb=0
PROGRAM DATA INPUT
40H
OEb=0
PROGRAM START
STATUS REGISTER READ
RESET COMMAND
PROGRAM FINISH
STANDBY
OUTPUT DISABLE
CEb=0
STATUS REGISTER CLEAR COMMAND
PROGRAM OR ERASE ERROR
CEb=0
ERROR
CEb=1
1ST COMMAND
RECOVERY READ SETUP
OEb=0, SC
RECOVERY READ
RESET COMMAND
CEb=0
1ST COMMAND
SA(1), SA(2)
OEb=0
RETRY PROGRAM SETUP
SECTOR ADDRESS INPUT
PROGRAM START
STATUS REGISTER READ
RESET COMMAND
PROGRAM FINISH
PROGRAM ERROR
OEb=0
STATUS REGISTER READ
OEb=1
OEb=0
STATUS REGISTER READ

FIG. 58

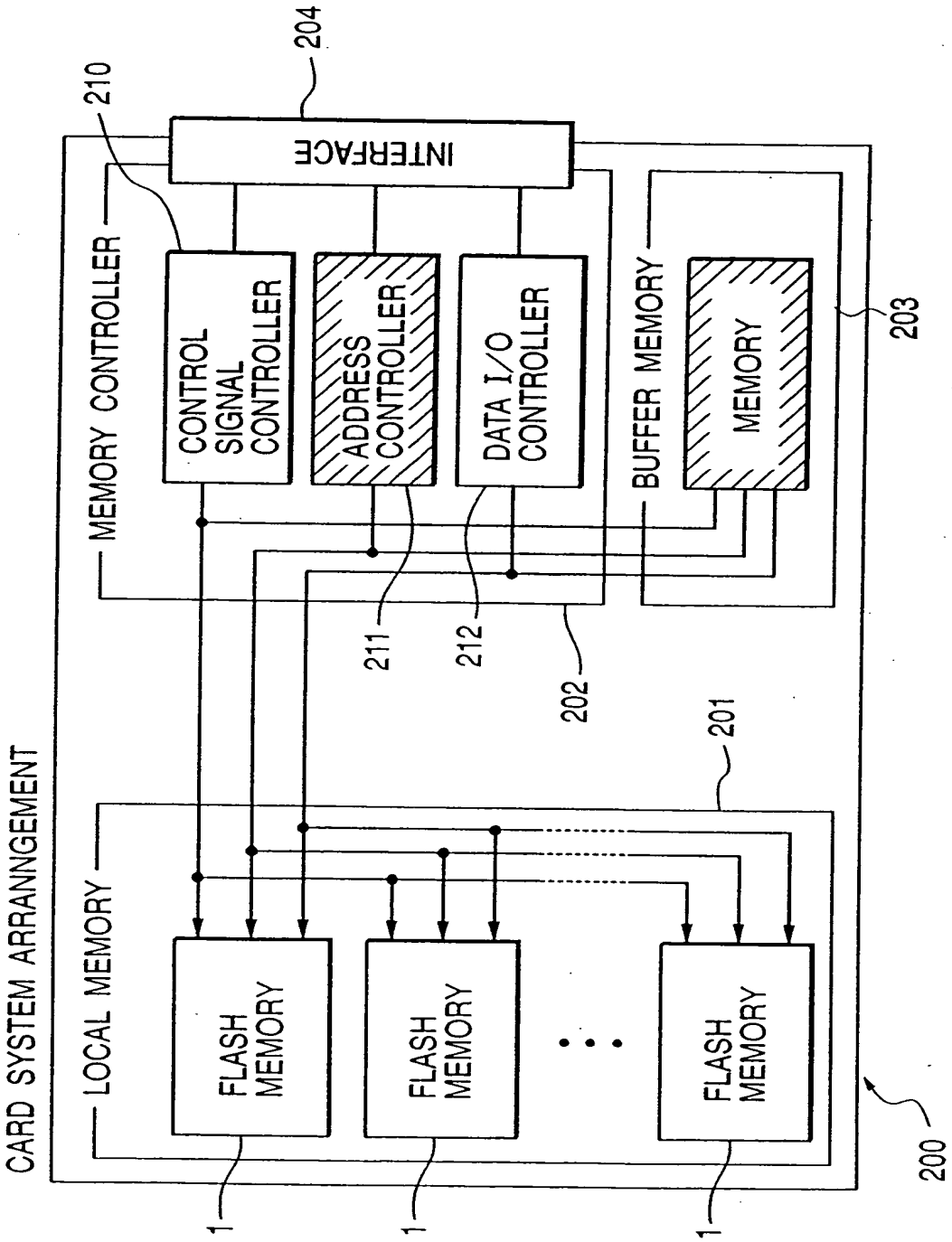

CARD SYSTEM ARRANGEMENT
LOCAL MEMORY
MEMORY CONTROLLER
BUFFER MEMORY
INTERFACE
CONTROL SIGNAL CONTROLLER
ADDRESS CONTROLLER
DATA I/O CONTROLLER
MEMORY
FLASH MEMORY
FLASH MEMORY
FLASH MEMORY
200
201
202
203
204
210
211
212
1
1
1

FIG. 59

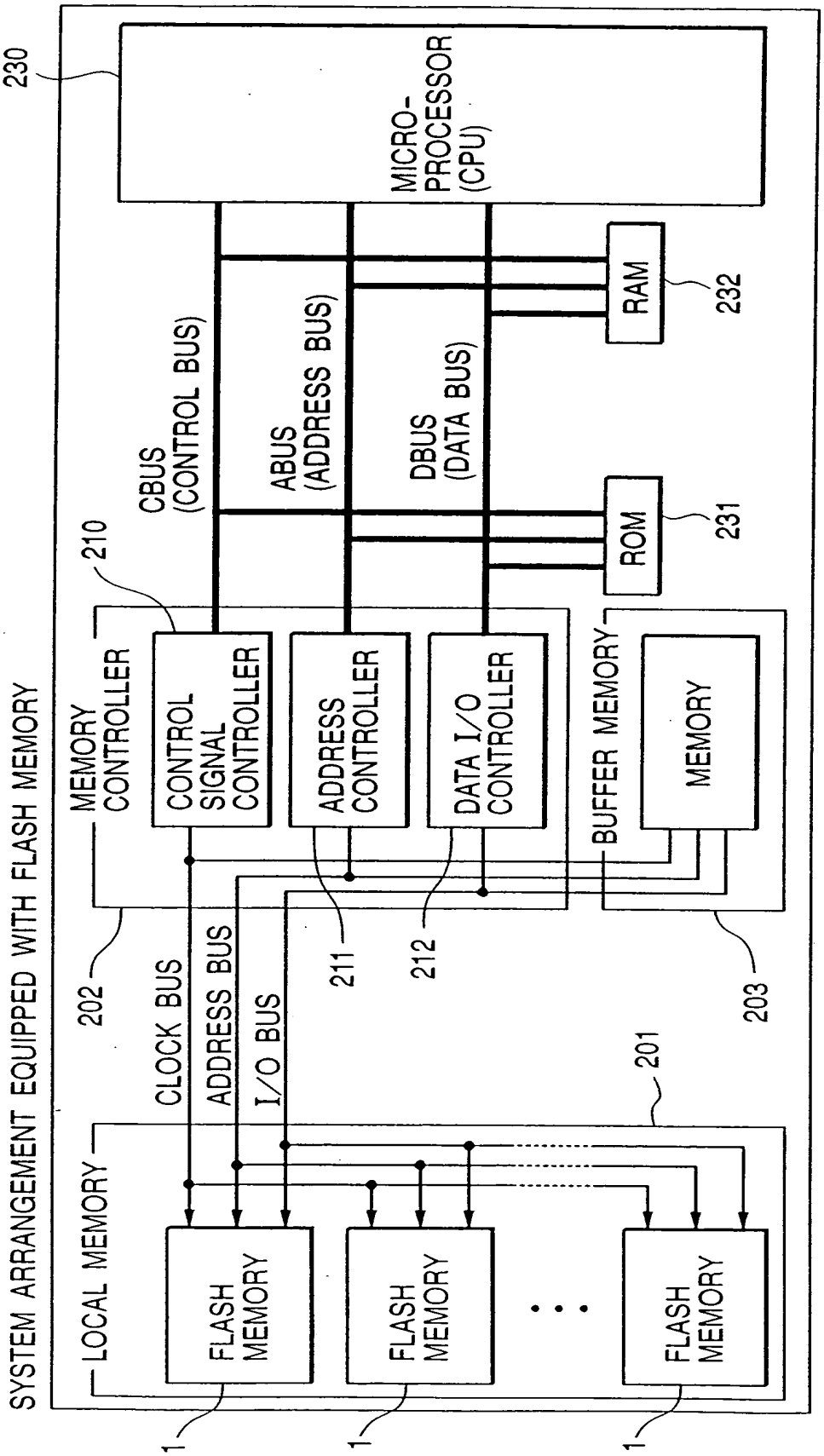

SYSTEM ARRANGEMENT EQUIPPED WITH FLASH MEMORY
230
MICRO-PROCESSOR (CPU)
RAM
232
ROM
231
CBUS (CONTROL BUS)
ABUS (ADDRESS BUS)
DBUS (DATA BUS)
210
MEMORY CONTROLLER
CONTROL SIGNAL CONTROLLER
ADDRESS CONTROLLER
DATA I/O CONTROLLER
BUFFER MEMORY
MEMORY
202
211
212
203
CLOCK BUS
ADDRESS BUS
I/O BUS
201
LOCAL MEMORY
FLASH MEMORY
FLASH MEMORY
FLASH MEMORY
1
1
1

FIG. 60

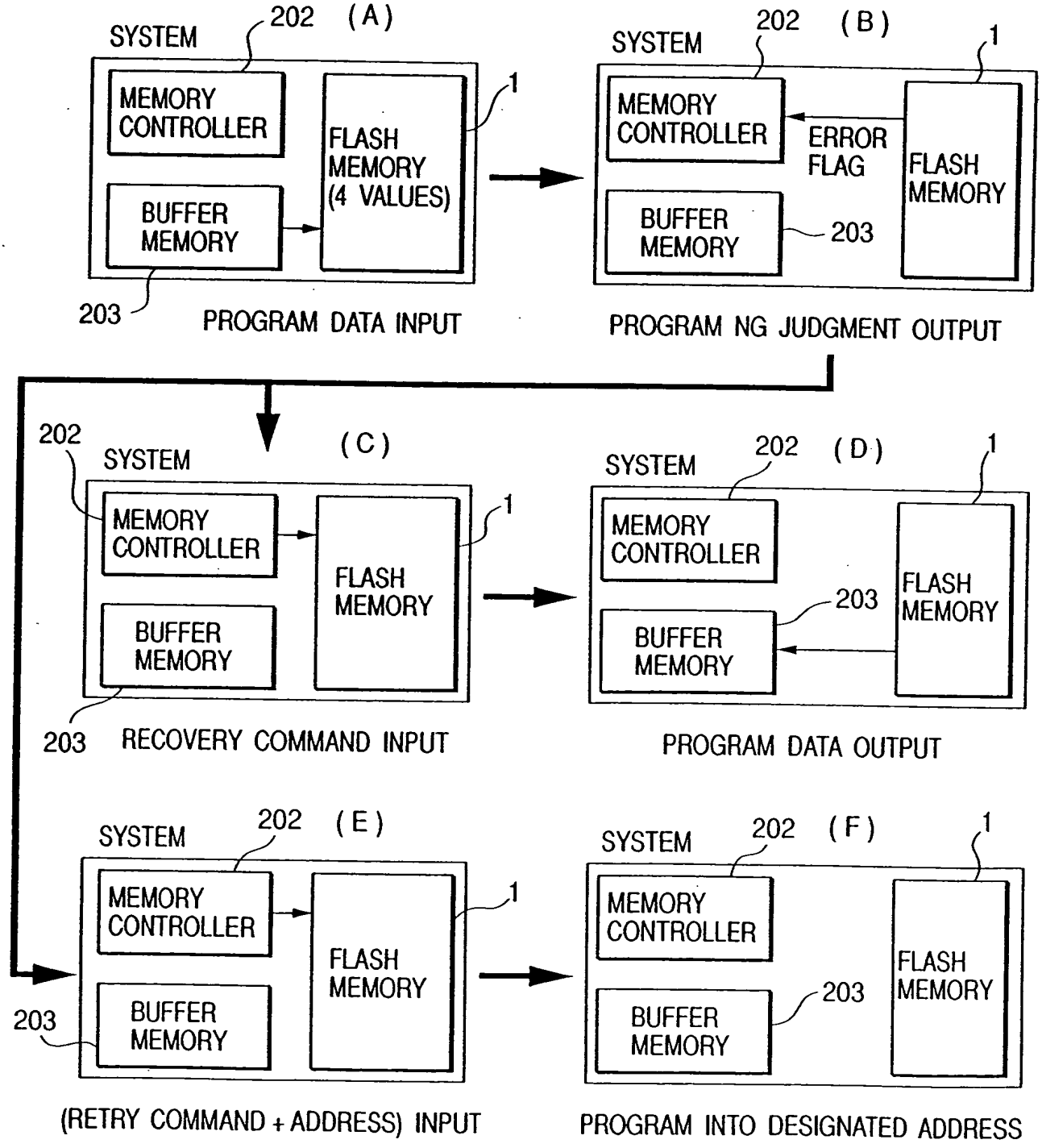
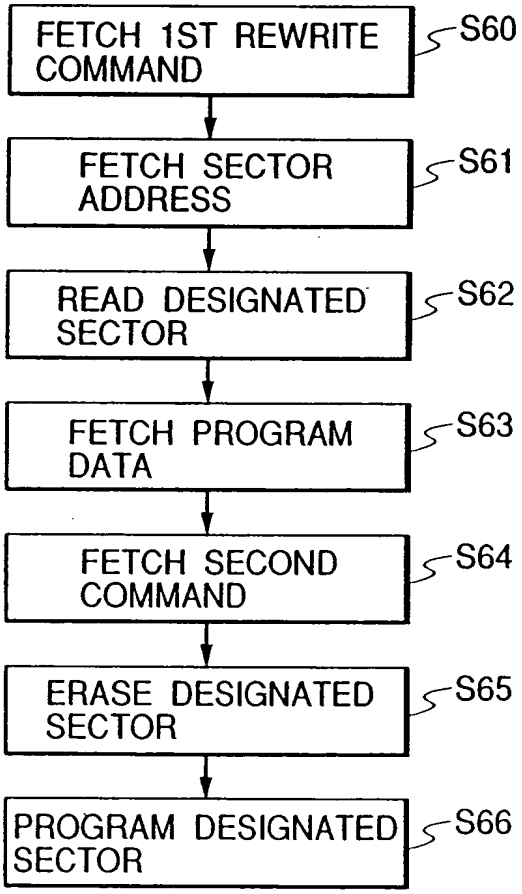

SYSTEM
202
(A)
MEMORY CONTROLLER
BUFFER MEMORY
FLASH MEMORY (4 VALUES)
1
203
PROGRAM DATA INPUT
SYSTEM
202
(B)
1
MEMORY CONTROLLER
ERROR FLAG
FLASH MEMORY
BUFFER MEMORY
203
PROGRAM NG JUDGMENT OUTPUT
202
SYSTEM
(C)
MEMORY CONTROLLER
BUFFER MEMORY
FLASH MEMORY
1
203
RECOVERY COMMAND INPUT
SYSTEM
202
(D)
1
MEMORY CONTROLLER
203
FLASH MEMORY
BUFFER MEMORY
PROGRAM DATA OUTPUT
SYSTEM
202
(E)
MEMORY CONTROLLER
FLASH MEMORY
1
203
BUFFER MEMORY
(RETRY COMMAND + ADDRESS) INPUT
SYSTEM
202
(F)
1
MEMORY CONTROLLER
FLASH MEMORY
203
BUFFER MEMORY
PROGRAM INTO DESIGNATED ADDRESS

FIG. 61



IN THE CASE THAT DATA OF ALL SECTORS ARE REWRITTEN

FIG. 62

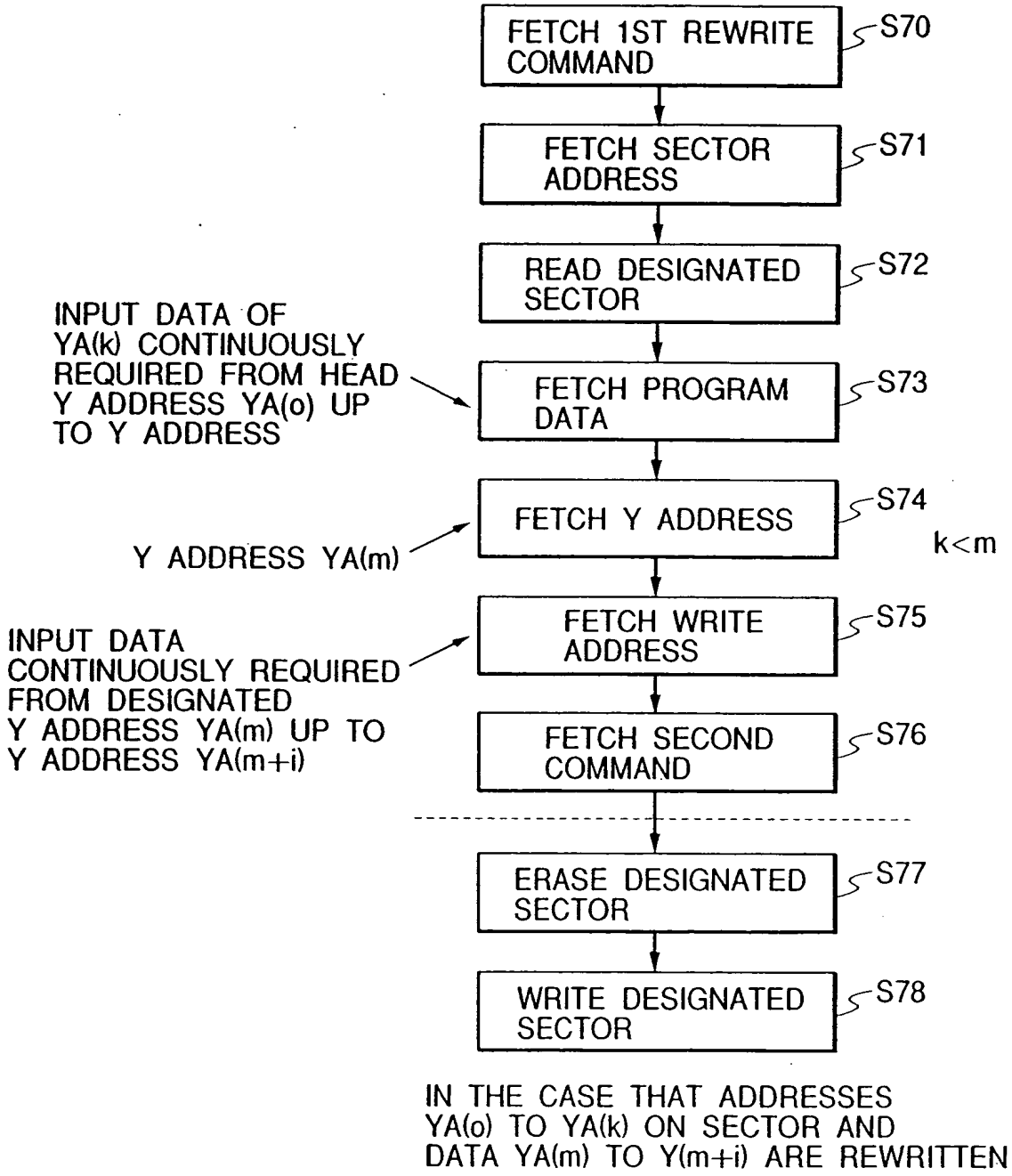

FETCH 1ST REWRITE COMMAND
S70
FETCH SECTOR ADDRESS
S71
READ DESIGNATED SECTOR
S72
INPUT DATA OF YA(k) CONTINUOUSLY REQUIRED FROM HEAD Y ADDRESS YA(o) UP TO Y ADDRESS
FETCH PROGRAM DATA
S73
FETCH Y ADDRESS
S74
k<m
Y ADDRESS YA(m)
FETCH WRITE ADDRESS
S75
INPUT DATA CONTINUOUSLY REQUIRED FROM DESIGNATED Y ADDRESS YA(m) UP TO Y ADDRESS YA(m+i)
FETCH SECOND COMMAND
S76
ERASE DESIGNATED SECTOR
S77
WRITE DESIGNATED SECTOR
S78
IN THE CASE THAT ADDRESSES YA(o) TO YA(k) ON SECTOR AND DATA YA(m) TO Y(m+i) ARE REWRITTEN

FIG. 63

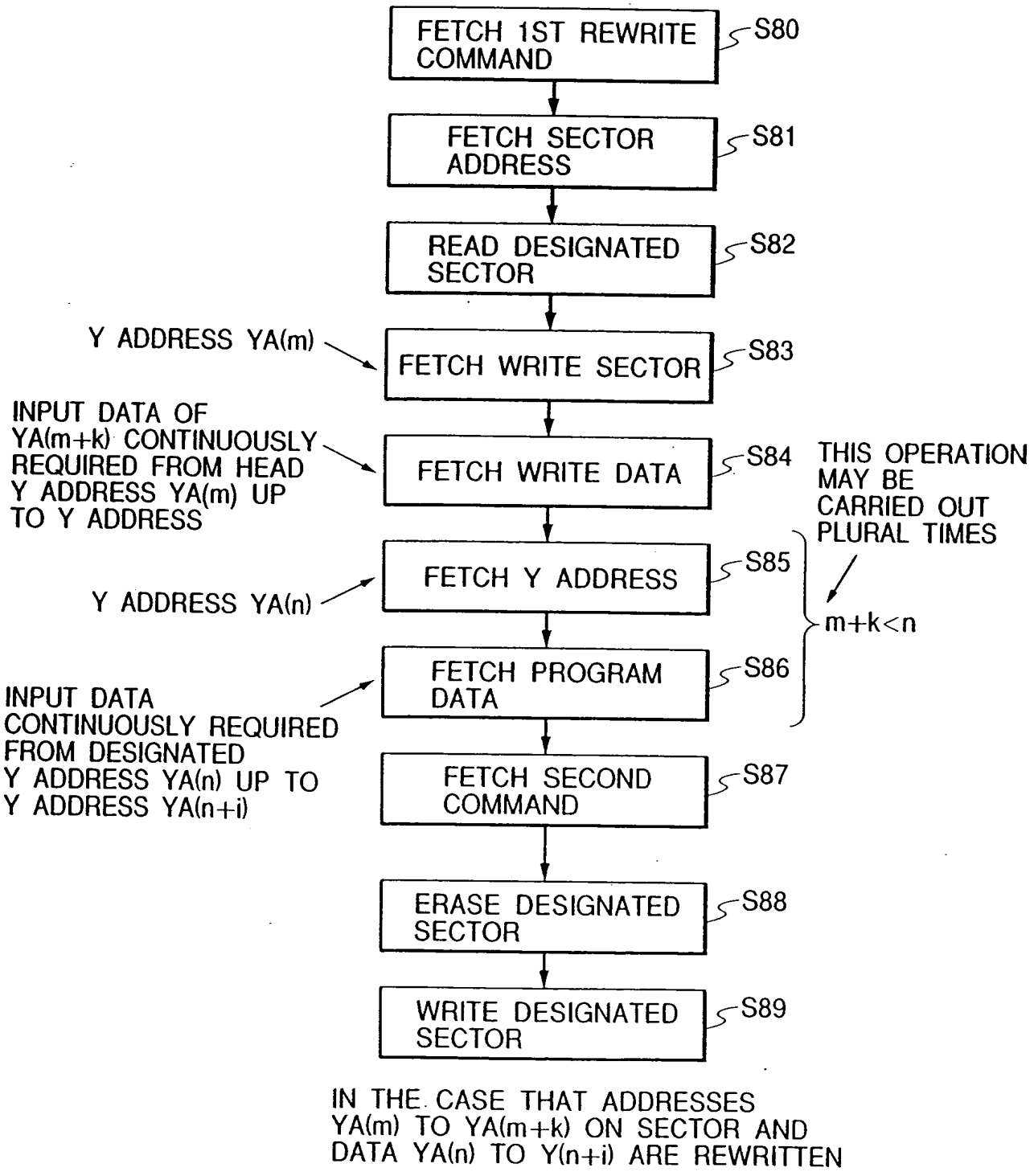

FETCH 1ST REWRITE COMMAND
S80
FETCH SECTOR ADDRESS
S81
READ DESIGNATED SECTOR
S82
Y ADDRESS YA(m)
FETCH WRITE SECTOR
S83
INPUT DATA OF YA(m+k) CONTINUOUSLY REQUIRED FROM HEAD Y ADDRESS YA(m) UP TO Y ADDRESS
FETCH WRITE DATA
S84
THIS OPERATION MAY BE CARRIED OUT PLURAL TIMES
FETCH Y ADDRESS
S85
Y ADDRESS YA(n)
m+k<n
FETCH PROGRAM DATA
S86
INPUT DATA CONTINUOUSLY REQUIRED FROM DESIGNATED Y ADDRESS YA(n) UP TO Y ADDRESS YA(n+i)
FETCH SECOND COMMAND
S87
ERASE DESIGNATED SECTOR
S88
WRITE DESIGNATED SECTOR
S89
IN THE CASE THAT ADDRESSES YA(m) TO YA(m+k) ON SECTOR AND DATA YA(n) TO Y(n+i) ARE REWRITTEN

FIG. 64

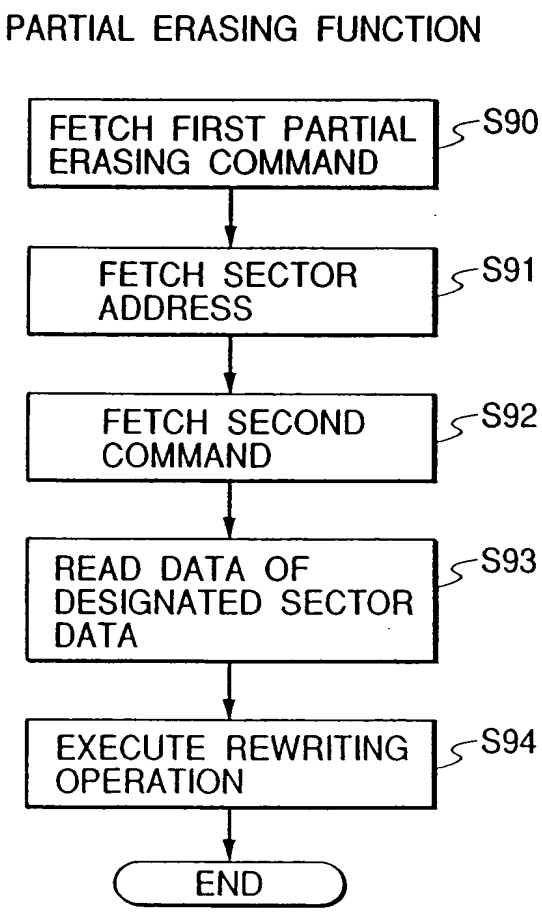
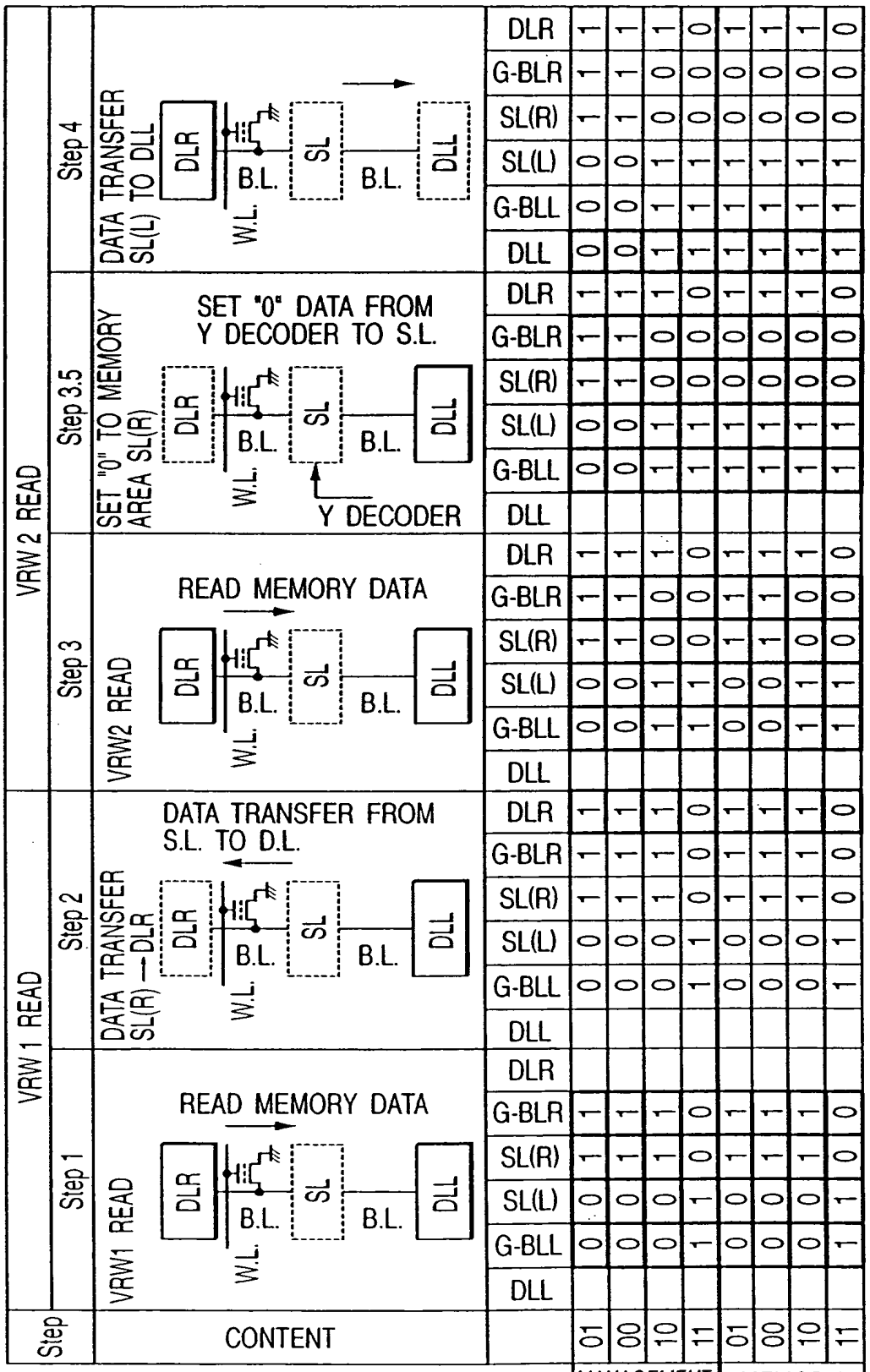

PARTIAL ERASING FUNCTION
FETCH FIRST PARTIAL ERASING COMMAND
S90
FETCH SECTOR ADDRESS
S91
FETCH SECOND COMMAND
S92
READ DATA OF DESIGNATED SECTOR DATA
S93
EXECUTE REWRITING OPERATION
S94
END

FIG. 65

※ SEQUENCE OPERATION WHEN RIGHT MAT IS SELECTED
'1' : IN THE CASE THAT POTENTIALS AT RESPECTIVE NODES ARE HIGH
'0' : IN THE CASE THAT POTENTIALS AT RESPECTIVE NODES ARE LOW



READ SEQUENCE WITHIN PARTIAL ERASE FLOW

FIG. 66

※ SEQUENCE OPERATION WHEN RIGHT MAT IS SELECTED
'1': IN THE CASE THAT POTENTIALS AT RESPECTIVE NODES ARE HIGH
'0': IN THE CASE THAT POTENTIALS AT RESPECTIVE NODES ARE LOW

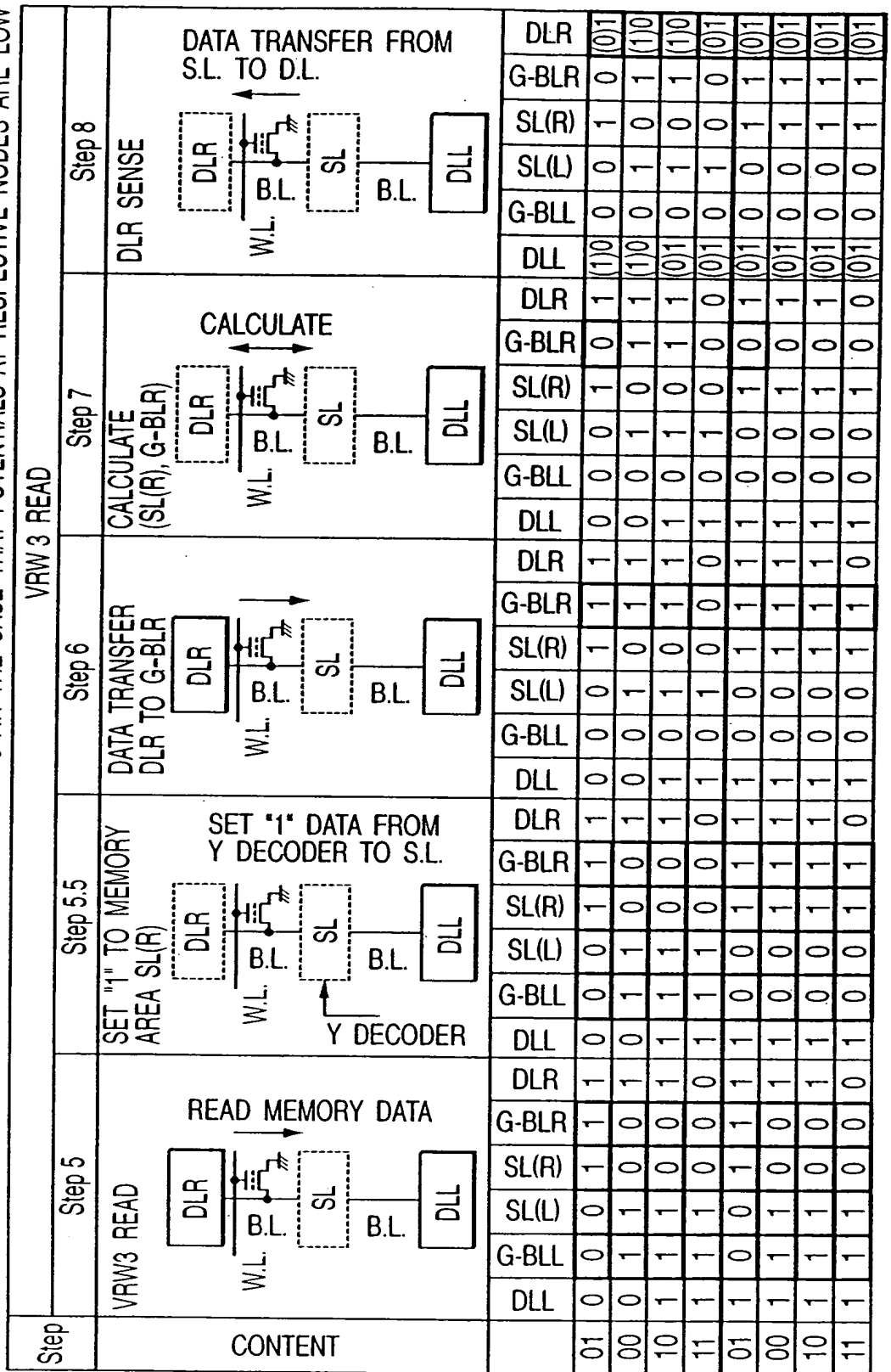


FIG. 67